FortiOS™ Handbook - Hardware Acceleration

VERSION 5.0

FORTINET DOCUMENT LIBRARY

http://docs.fortinet.com

FORTINET VIDEO GUIDE

http://video.fortinet.com

FORTINET BLOG

https://blog.fortinet.com

CUSTOMER SERVICE & SUPPORT

https://support.fortinet.com

FORTIGATE COOKBOOK

http://cookbook.fortinet.com

FORTINET TRAINING SERVICES

http://www.fortinet.com/training

FORTIGUARD CENTER

http://www.fortiguard.com

END USER LICENSE AGREEMENT

http://www.fortinet.com/doc/legal/EULA.pdf

FEEDBACK

Email: techdocs@fortinet.com

Wednesday, March 04, 2015

FortiOS™ Handbook - Hardware Acceleration

01-500-112804-20150304

TABLE OF CONTENTS

Change Log **6**
Introduction **7**
Hardware acceleration overview **8**
Content processors (CP4, CP5, CP6 and CP8) 8
CP8 capabilities 8
CP6 capabilities 8
CP5 capabilities 9
CP4 capabilities 9
Determining the content processor in your FortiGate unit 9
Viewing SSL acceleration status 10
Disabling CP offloading for firewall policies 10
Security processors (SPs) 10
SP Processing Flow 11
Displaying information about security processing modules 13
Network processors (NP1, NP2, NP3, NP4 and NP6) 13
Determining the network processors installed on your FortiGate unit 14
How NP hardware acceleration alters packet flow 14
NP processors and traffic logging and monitoring 15
NP session offloading in HA active-active configuration 16
Configuring NP HMAC check offloading 16
Software switch interfaces and NP processors 16
Configuring NP accelerated VPN encryption/decryption offloading 16
Disabling NP acceleration for individual IPsec VPN phase 1s 17
Disabling NP offloading for unsupported IPsec encryption or authentication algorithms 17
Disabling NP offloading for firewall policies 18
Checking that traffic is offloaded by NP processors 18
Using the packet sniffer 18
Checking the firewall session offload tag 18
Verifying IPsec VPN traffic offloading 19
Controlling IPS NPx and CPx acceleration 20
Dedicated Management CPU 20
NP6 Acceleration **21**
NP6 session fast path requirements 21
Packet fast path requirements 22

Mixing fast path and non-fast path traffic 22
Viewing your FortiGate NP6 processor configuration 23
Increasing NP6 offloading capacity using link aggregation groups (LAGs) 23
Configuring Inter-VDOM link acceleration with NP6 processors 23
Using VLANs to add more accelerated Inter-VDOM links 24
Confirm that the traffic is accelerated 25
Configuring individual NP6 processors 26
Enabling per-session accounting for offloaded NP6 sessions 31
FortiGate NP6 architectures 32
FortiGate-500D fast path architecture 32
FortiGate-1000D fast path architecture 33
FortiGate-1200D fast path architecture 35
FortiGate-1500D fast path architecture 37
FortiGate-3200D fast path architecture 39
FortiGate-3700D fast path architecture 41
FortiGate-3700D low latency fast path architecture 41
FortiGate-3700D normal latency fast path architecture 42
FortiGate-3810D fast path architecture 45
FortiGate-5001D fast path architecture 46
NP4 Acceleration 49
Viewing your FortiGate's NP4 configuration 49
NP4lite CLI commands (disabling NP4Lite offloading) 50
Configuring NP4 traffic offloading 50
NP4 session fast path requirements 50
Packet fast path requirements 51
Mixing fast path and non-fast path traffic 51
NP4 traffic shaping offloading 51
NP4 IPsec VPN offloading 52
NP4 IPsec VPN offloading configuration example 52
Accelerated policy mode IPsec configuration 54
Accelerated interface mode IPsec configuration 55
Configuring Inter-VDOM link acceleration with NP4 processors 56
Using VLANs to add more accelerated Inter-VDOM links 58
Confirm that the traffic is accelerated 59
Offloading NP4 anomaly detection 60
FortiGate NP4 architectures 63
FortiGate-600C 63
FortiGate-800C 64
FortiGate-1000C 65
FortiGate-1240B 66
FortiGate-3040B 67
FortiGate-3140B 68

FortiGate-3140B — load balance mode 69
FortiGate-3240C 70
FortiGate-3600C 71
XAUI interfaces 72
FortiGate-3950B and FortiGate-3951B 72
FortiGate-3950B and FortiGate-3951B — load balance mode 74
FortiGate-5001C 75
FortiGate-5001B 76
Setting switch-mode mapping on the ADM-XD4 77

Change Log

Date	Change Description
4 March 2015	Updated 5.0 version.

Introduction

This document describes the hardware components that Fortinet builds into FortiGate devices to accelerate traffic through FortiGate units. Three types of hardware acceleration components are described:

- Content processors (CPs) that accelerate a wide range of security functions
- Security processors (SPs) that accelerate specific security functions
- Network processors (NPs) that offload network traffic to specialized hardware that is optimized to provide high levels of network throughput.

This FortiOS Handbook chapter contains the following sections:

Hardware acceleration overview describes the capabilities of FortiGate content processors (CPs), security processors (SPs) and network processors (NPs). This chapter also describes how to determine the hardware acceleration components installed in your FortiGate unit and contains some configuration details and examples.

NP6 Acceleration describes the FortiGate NP6 network processor.

FortiGate NP6 architectures contains details about the network processing architectures of FortiGate units that contain NP6 processors.

NP4 Acceleration describes the FortiGate NP4 network processor.

FortiGate NP4 architectures contains details about the network processing architectures of FortiGate units that contain NP4 processors.

Hardware acceleration overview

Most FortiGate models have specialized acceleration hardware that can offload resource intensive processing from main processing (CPU) resources. Most FortiGate units include specialized content processors (CPs) that accelerate a wide range of important security processes such as virus scanning, attack detection, encryption and decryption. (Only selected entry-level FortiGate models do not include a CP processor.) Many FortiGate models also contain security processors (SPs) that accelerate processing for specific security features such as IPS and network processors (NPs) that offload processing of high volume network traffic.

Content processors (CP4, CP5, CP6 and CP8)

Most FortiGate models contain FortiASIC Content Processors (CPs) that accelerate many common resource intensive security related processes. CPs work at the system level with tasks being offloaded to them as determined by the main CPU. Capabilities of the CPs vary by model. Newer FortiGate units include CP8 processors. Older CP versions still in use in currently operating FortiGate models include the CP4, CP5, and CP6.

CP8 capabilities

The CP8 content processor provides the following services:

- IPS signature matching acceleration
- High performance VPN bulk data engine
  - IPsec and SSL/TLS protocol processor
  - DES/3DES/AES in accordance with FIPS46-3/FIPS81/FIPS197
  - ARC4 in compliance with RC4
  - MD5/SHA-1/SHA256 with RFC1321 and FIPS180
  - HMAC in accordance with RFC2104/2403/2404 and FIPS198
- Key Exchange Processor support high performance IKE and RSA computation
  - Public key exponentiation engine with hardware CRT support
  - Primarily checking for RSA key generation
  - Handshake accelerator with automatic key material generation
  - Random Number generator compliance with ANSI X9.31
  - Sub public key engine (PKCE) to support up to 4094 bit operation directly
- Message authentication module offers high performance cryptographic engine for calculating SHA256/SHA1/MD5 of data up to 4G bytes (used by many applications)
- PCI express Gen 2 four lanes interface
- Cascade Interface for chip expansion

CP6 capabilities

- Dual content processors
- FIPS-compliant DES/3DES/AES encryption and decryption
- SHA-1 and MD5 HMAC with RFC1321 and FIPS180

- HMAC in accordance with RFC2104/2403/2404 and FIPS198
- IPsec protocol processor
- High performance IPsec engine
- Random Number generator compliance with ANSI X9.31
- Key exchange processor for high performance IKE and RSA computation
- Script Processor
- SSL/TLS protocol processor for SSL content scanning and SSL acceleration

CP5 capabilities

- FIPS-compliant DES/3DES/AES encryption and decryption
- SHA-1 and MD5 HMAC with RFC1321/2104/2403/2404 and FIPS180/FIPS198
- IPsec protocol processor
- High performance IPSEC Engine
- Random Number generator compliant with ANSI X9.31
- Public Key Crypto Engine supports high performance IKE and RSA computation
- Script Processor

CP4 capabilities

- FIPS-compliant DES/3DES/AES encryption and decryption
- SHA-1 and MD5 HMAC
- IPSEC protocol processor
- Random Number generator
- Public Key Crypto Engine
- Content processing engine
- ANSI X9.31 and PKCS#1 certificate support

Determining the content processor in your FortiGate unit

Use the `get hardware status` CLI command to determine which content processor your FortiGate unit contains. The output looks like this:

```
get hardware status
Model name: FortiGate-100D
ASIC version: CP8
ASIC SRAM: 64M
CPU: Intel(R) Atom(TM) CPU D525 @ 1.80GHz
Number of CPUs: 4
RAM: 1977 MB
Compact Flash: 15331 MB /dev/sda
Hard disk: 15272 MB /dev/sda
USB Flash: not available
Network Card chipset: Intel(R) PRO/1000 Network Connection (rev.0000)
Network Card chipset: bcm-sw Ethernet driver 1.0 (rev.)
```

The ASIC version line lists the content processor model number.

Viewing SSL acceleration status

You can view the status of SSL acceleration using the following command:

```
get vpn status ssl hw-acceleration-status
Acceleration hardware detected: kxp=on cipher=on
```

Disabling CP offloading for firewall policies

If you want to completely disable offloading to CP processors for test purposes or other reasons, you can do so in security policies. Here are some examples:

For IPv4 security policies.

```
config firewall policy
   edit 1
      set auto-asic-offload disable
   end
```

For IPv6 security policies.

```
config firewall policy6
   edit 1
      set auto-asic-offload disable
   end
```

For multicast security policies.

```
config firewall multicast-policy
   edit 1
      set auto-asic-offload disable
   end
```

Disabling `auto-asic-offload` also disables NP offloading.

Security processors (SPs)

FortiGate Security Processing (SP) modules, such as the SP3 but also including the XLP, XG2, XE2, FE8, and CE4, work at both the interface and system level to increase overall system performance by accelerating specialized security processing. You can configure the SP to favor IPS over firewall processing in hostile high-traffic environments.

SP processors include their own IPS engine which is similar to the FortiOS IPS engine but with the following limitations:

- The SP IPS engine does not support SSL deep inspection. When you have SSL deep inspection enabled for a security policy that includes flow-based inspection or IPS, offloading to the SP is disabled and traffic is processed by the FortiGate CPU and CP processors.

- The SP IPS engine does not support FortiGuard Web Filtering. When you enable flow-based FortiGuard Web Filtering on a FortiGate unit with an SP processor, the SP processor cannot perform FortiGuard lookups and web pages fail to load.

The following security processors are available:

- The SP3 (XLP) is built into the FortiGate-5101B and provides IPS acceleration. No special configuration is required. All IPS processing, including traffic accepted by IPv4 and IPv6 traffic policies and IPv4 and IPv6 DoS policies is accelerated by the built-in SP3 processors.
- The FMC-XG2 is an FMC module with two 10Gb/s SPF+ interfaces that can be used on FortiGate-3950B and FortiGate-3951B units.
- The FortiGate-3140B also contains a built-in XG2 using ports 19 and 20.
- The ADM-XE2 is a dual-width AMC module with two 10Gb/s interfaces that can be used on FortiGate-3810A and FortiGate-5001A-DW systems.
- The ADM-FE8 is a dual-width AMC module with eight 1Gb/s interfaces that can be used with the FortiGate-3810A.
- The ASM-CE4 is a single-width AMC module with four 10/100/1000 Mb/s interfaces that can be used on FortiGate-3016B and FortiGate-3810A units.

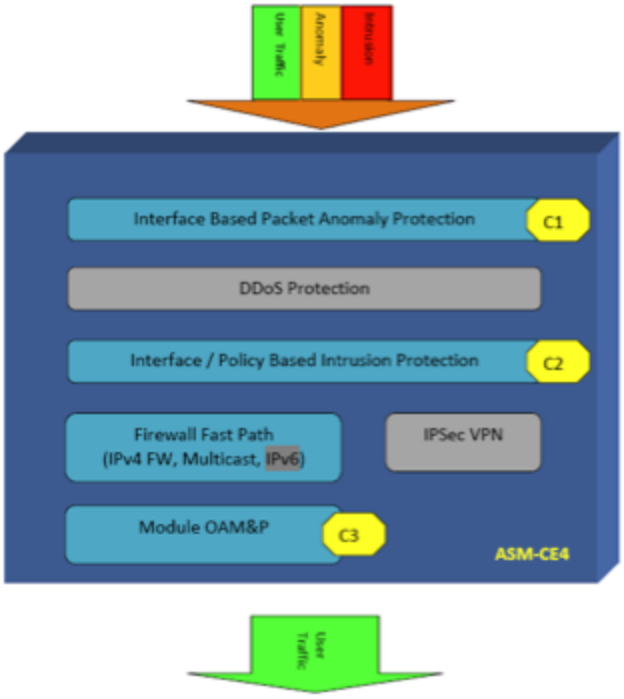
SP Processing Flow

SP processors provide an integrated high performance fast path multilayer solution for both intrusion protection and firewall functions. The multilayered protection starts from anomaly checking at packet level to ensure each packet is sound and reasonable. Immediately after that, a sophisticated set of interface based packet anomaly protection, DDoS protection, policy based intrusion protection, firewall fast path, and behavior based methods are employed to prevent DDoS attacks from the rest of system.

Then the packets enter an interface/policy based intrusion protection system,

where each packet is evaluated against a set of signatures. The end result is streams of user packets that are free of anomaly and attacks, entering the fast path system for unicast or multicast fast path forwarding.

SP processing flow



Displaying information about security processing modules

You can display information about installed SP modules using the CLI command

```
diagnose npu spm
```

For example, for the FortiGate-5101C:

```
FG-5101C # diagnose npu spm list
Available SP Modules:

ID Model      Slot     Interface
0 xh0         built-in  port1, port2, port3, port4,
                        base1, base2, fabric1, fabric2
                        eth10, eth11, eth12, eth13
                        eth14, eth15, eth16, eth17
                        eth18, eth19
```

You can also use this command to get more info about SP processing. This example shows how to display details about how the module is processing sessions using the syn proxy.

```
diagnose npu spm dos synproxy <sp_id>
```

This is a partial output of the command:

```
Number of proxied TCP connections :                 0
Number of working proxied TCP connections :         0
Number of retired TCP connections :                 0
Number of valid TCP connections :                   0
Number of attacks, no ACK from client :             0
Number of no SYN-ACK from server :                  0
Number of reset by server (service not supportted): 0
Number of establised session timeout :              0
Client timeout setting :                            3 Seconds
Server timeout setting :                            3 Seconds
```

Network processors (NP1, NP2, NP3, NP4 and NP6)

FortiASIC network processors work at the interface level to accelerate traffic by offloading traffic from the main CPU. Current models contain NP4 and NP6 network processors. Older FortiGate models include NP1 network processors (also known as FortiAccel, or FA2) and NP2 network processors.

The traffic that can be offloaded, maximum throughput, and number of network interfaces supported by each varies by processor model:

- NP6 supports offloading of most IPv4 and IPv6 traffic, IPsec VPN encryption, CAPWAP traffic, and multicast traffic. The NP6 has a capacity of 40 Gbps through 4 x 10 Gbps interfaces or 3 x 10 Gbps and 16 x 1 Gbps interfaces. For details about the NP6 processor, see NP6 Acceleration on page 21 and for information about FortiGate models with NP6 processors, see FortiGate NP6 architectures on page 32.
- NP4 supports offloading of most IPv4 firewall traffic and IPsec VPN encryption. The NP4 has a capacity of 20 Gbps through 2 x 10 Gbps interfaces. For details about NP4 processors, see NP4 Acceleration on page 49 and for information about FortiGate models with NP4 processors, see FortiGate NP4 architectures on page 63.
- NP2 supports IPv4 firewall and IPsec VPN acceleration. The NP2 has a capacity of 2 Gbps through 2 x 10 Gbps interfaces or 4 x 1 Gbps interfaces.

- NP1 supports IPv4 firewall and IPsec VPN acceleration with 2 Gbps capacity. The NP1 has a capacity of 2 Gbps through 2 x 1 Gbps interfaces.
  - The NP1 does not support frames greater than 1500 bytes. If your network uses jumbo frames, you may need to adjust the MTU (Maximum Transmission Unit) of devices connected to NP1 ports. Maximum frame size for NP2, NP4, and NP6 processors is 9000 bytes.
  - For both NP1 and NP2 network processors, ports attached to a network processor cannot be used for firmware installation by TFTP.

Sessions that require proxy-based and flow based security features (for example, virus scanning, IPS, application control and so on) are not fast pathed and must be processed by the CPU

Determining the network processors installed on your FortiGate unit

Use the following command to list the NP6 processors in your FortiGate unit:

```
diagnose npu np6 port-list
```

To list other network processors on your FortiGate unit, use the following CLI command.

```
get hardware npu <model> list
```

`<model>` can be `legacy`, `np1`, `np2` or `np4`.

The output lists the interfaces that have the specified processor. For example, for a FortiGate-5001B:

```
get hardware npu np4 list
ID    Model          Slot       Interface
0     On-board                  port1 port2 port3 port4
                                fabric1 base1 npu0-vlink0 npu0-vlink1
1     On-board                  port5 port6 port7 port8
                                fabric2 base2 npu1-vlink0 npu1-vlink1
```

The `npu0-vlink0`, `npu1-vlink1` etc interfaces are used for accelerating inter-VDOM links.

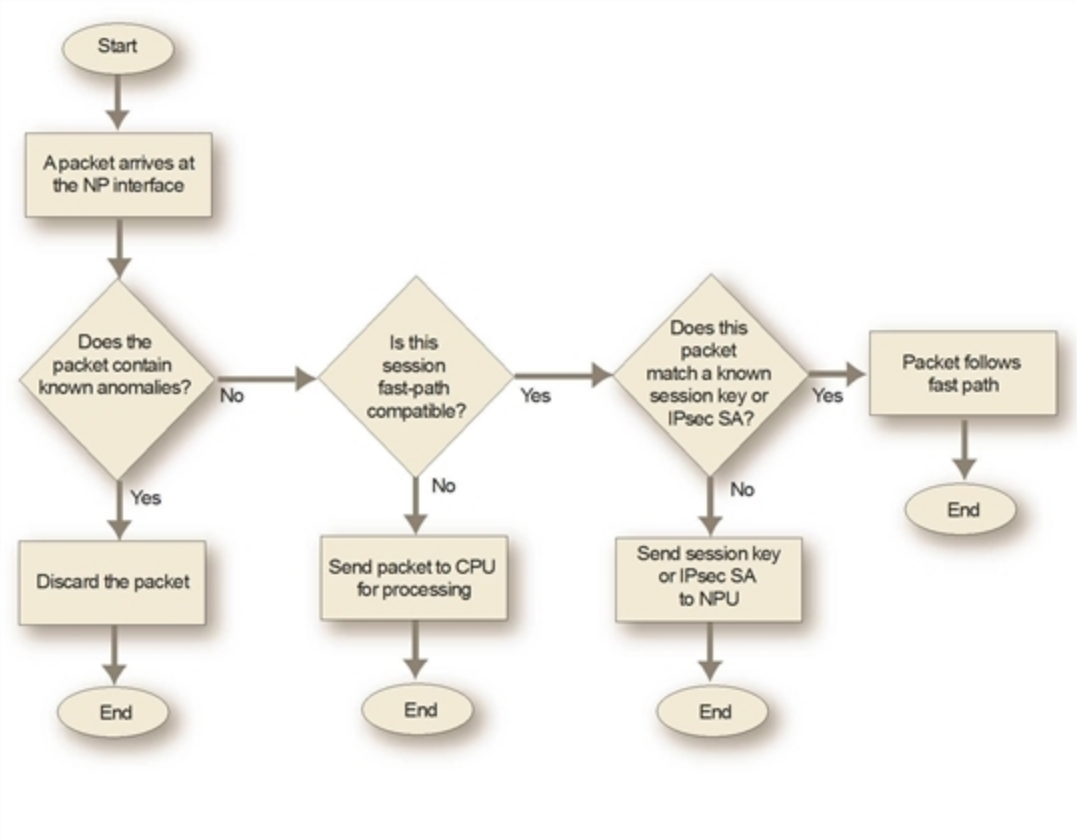
How NP hardware acceleration alters packet flow

NP hardware acceleration generally alters packet flow as follows:

1. Packets initiating a session pass to the FortiGate unit's main processing resources (CPU).
2. The FortiGate unit assesses whether the session matches fast path (offload) requirements.
   To be suitable for offloading, traffic must possess only characteristics that can be processed by the fast path. The list of requirements depends on the processor, see NP6 session fast path requirements on page 21 or NP4 session fast path requirements on page 50.
   If the session can be fast pathed, the FortiGate unit sends the session key or IPsec security association (SA) and configured firewall processing action to the appropriate network processor.
3. Network processors continuously match packets arriving on their attached ports against the session keys and SAs they have received.
   - If a network processor's network interface is configured to perform hardware accelerated anomaly checks, the network processor drops or accepts packets that match the configured anomaly patterns. These checks are separate from and in advance of anomaly checks performed by IPS, which is not compatible with network processor offloading. See Offloading NP4 anomaly detection on page 60.

- The network processor next checks for a matching session key or SA. If a matching session key or SA is found, and if the packet meets packet requirements, the network processor processes the packet according to the configured action and then sends the resulting packet. This is the actual offloading step. Performing this processing on the NP processor improves overall performance because the NP processor is optimized for this task. As well, overall FortiGate performance is improved because the CPU has fewer sessions to process.

NP network processor packet flow



- If a matching session key or SA is not found, or if the packet does not meet packet requirements, the packet cannot be offloaded. The network processor sends the data to the FortiGate unit's CPU, which processes the packet.

Encryption and decryption of IPsec traffic originating from the FortiGate can utilize network processor encryption capabilities.

Packet forwarding rates vary by the percentage of offloadable processing and the type of network processing required by your configuration, but are independent of frame size. For optimal traffic types, network throughput can equal wire speed.

NP processors and traffic logging and monitoring

Except for the NP6, network processors do not count offloaded packets, and offloaded packets are not logged by traffic logging and are not included in traffic statistics and traffic log reports.

NP6 processors support per-session traffic and byte counters, Ethernet MIB matching, and reporting through messages resulting in traffic statistics and traffic log reporting.

NP session offloading in HA active-active configuration

Network processors can improve network performance in active-active (load balancing) high availability (HA) configurations, even though traffic deviates from general offloading patterns, involving more than one network processor, each in a separate FortiGate unit. No additional offloading requirements apply.

Once the primary FortiGate unit's main processing resources send a session key to its network processor(s), network processor(s) on the primary unit can redirect any subsequent session traffic to other cluster members, reducing traffic redirection load on the primary unit's main processing resources.

As subordinate units receive redirected traffic, each network processor in the cluster assesses and processes session offloading independently from the primary unit. Session key states of each network processor are not part of synchronization traffic between HA members.

Configuring NP HMAC check offloading

Hash-based Message Authentication Code (HMAC) checks offloaded to network processors by default. You can enter the following command to disable this feature:

```
configure system global
   set ipsec-hmac-offload disable
end
```

Software switch interfaces and NP processors

FortiOS supports creating a software switch by grouping two or more FortiGate physical interfaces into a single virtual or software switch interface. All of the interfaces in this virtual switch act like interfaces in a hardware switch in that they all have the same IP address and can be connected to the same network. You create a software switch interface from the CLI using the command `config system switch-interface`.

The software switch is a bridge group of several interfaces, and the FortiGate CPU maintains the mac-port table for this bridge. As a result of this CPU involvement, traffic processed by a software switch interface is not offloaded to network processors.

Configuring NP accelerated VPN encryption/decryption offloading

Network processing unit (`npu`) settings configure offloading behavior for IPsec VPN. Configured behavior applies to all network processors contained by the FortiGate unit itself or any installed AMC modules.

```
config system npu
   set enc-offload-antireplay {enable | disable}
   set dec-offload-antireplay {enable | disable}
   set offload-ipsec-host {enable | disable}
end
```

Variables	Description	Default
`enc-offload-antireplay {enable | disable}`	Enable or disable offloading of IPsec encryption. This option is used only when replay detection is enabled in Phase 2 configuration. If replay detection is disabled, encryption is always offloaded.	disable
`dec-offload-antireplay {enable | disable}`	Enable or disable offloading of IPsec decryption. This option is used only when replay detection is enabled in Phase 2 configuration. If replay detection is disabled, decryption is always offloaded.	enable
`offload-ipsec-host {enable | disable}`	Enable or disable offloading of IPsec encryption of traffic from local host (FortiGate unit). **Note:** For this option to take effect, the FortiGate unit must have previously sent the security association (SA) to the network processor.	disable

Example

You could configure the offloading of encryption and decryption for an IPsec SA that was sent to the network processor.

```
config system npu
   set enc-offload-antireplay enable
   set dec-offload-antireplay enable
   set offload-ipsec-host enable
end
```

Disabling NP acceleration for individual IPsec VPN phase 1s

Use the following command to disable NP offloading for an interface-based IPsec VPN phase 1:

```
config vpn ipsec phase1-interface
   edit phase-1-name
      set npu-offload disable
end
```

Use the following command to disable NP offloading for a policy-based IPsec VPN phase 1:

```
config vpn ipsec phase1
   edit phase-1-name
      set npu-offload disable
end
```

The `npu-offload` option is enabled by default.

Disabling NP offloading for unsupported IPsec encryption or authentication algorithms

In general, more recent IPsec VPN encryption and authentication algorithms may not be supported by older NP processors. For example, NP4 network processors do not support SHA-256, SHA-384, and SHA-512. IPsec traffic with unsupported algorithms is not offloaded and instead is processed by the FortiGate CPU. In addition, this

configuration may cause packet loss and other performance issues. If you experience packet loss or performance problems you should set the `npu-offload` option to `disable`. Future FortiOS versions should prevent selecting algorithms not supported by the hardware.

Disabling NP offloading for firewall policies

Use the following options to disable NP offloading for specific security policies:

For IPv4 security policies.

```
config firewall policy
   edit 1
      set auto-asic-offload disable
   end
```

For IPv6 security policies.

```
config firewall policy6
   edit 1
      set auto-asic-offload disable
   end
```

For multicast security policies.

```
config firewall multicast-policy
   edit 1
      set auto-asic-offload disable
   end
```

Checking that traffic is offloaded by NP processors

A number of diagnose commands can be used to verify that traffic is being offloaded.

Using the packet sniffer

Use the packet sniffer to verify that traffic is offloaded. Offloaded traffic is not picked up by the packet sniffer so if you are sending traffic through the FortiGate unit and it is not showing up on the packet sniffer you can conclude that it is offloaded.

```
diag sniffer packet port1 <option>
```

Checking the firewall session offload tag

Use the `diagnose sys session list` command to display sessions. If the output for a session includes the `npu info` field you should see information about session being offloaded. If the output doesn't contain an `npu info` field then the session has not been offloaded.

```
diagnose sys session list
session info: proto=6 proto_state=01 duration=34 expire=3565 timeout=3600
    flags=00000000 sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=295/3/1 reply=60/1/1 tuples=2
```

```
orgin->sink: org pre->post, reply pre->post dev=48->6/6->48 gwy=10.1.100.11/11.11.11.1
hook=pre dir=org act=noop 172.16.200.55:56453->10.1.100.11:80(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.11:80->172.16.200.55:56453(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 id_policy_id=0 auth_info=0 chk_client_info=0 vd=4
serial=0000091c tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=172.16.200.55, bps=393
npu_state=00000000
npu info: flag=0x81/0x81, offload=4/4, ips_offload=0/0, epid=1/23, ipid=23/1,
    vlan=32779/0
```

Verifying IPsec VPN traffic offloading

The following commands can be used to verify IPsec VPN traffic offloading to NP processors.

```
diagnose vpn ipsec status
NP1/NP2/NP4_0/sp_0_0:
      null: 0 0
      des: 0 0
       3des: 4075 4074
      aes: 0 0
      aria: 0 0
      seed: 0 0
      null: 0 0
       md5: 4075 4074
      sha1: 0 0
      sha256: 0 0
      sha384: 0 0
      sha512: 0 0

diagnose vpn tunnel list
list all ipsec tunnel in vd 3
------------------------------------------------------
name=p1-vdom1 ver=1 serial=5 11.11.11.1:0->11.11.11.2:0 lgwy=static tun=tunnel
    mode=auto bound_if=47
proxyid_num=1 child_num=0 refcnt=8 ilast=2 olast=2
stat: rxp=3076 txp=1667 rxb=4299623276 txb=66323
dpd: mode=active on=1 idle=5000ms retry=3 count=0 seqno=20
natt: mode=none draft=0 interval=0 remote_port=0
proxyid=p2-vdom1 proto=0 sa=1 ref=2 auto_negotiate=0 serial=1
src: 0:0.0.0.0/0.0.0.0:0
dst: 0:0.0.0.0/0.0.0.0:0
SA: ref=6 options=0000000e type=00 soft=0 mtu=1436 expire=1736 replaywin=2048 seqno=680
life: type=01 bytes=0/0 timeout=1748/1800
dec: spi=ae01010c esp=3des key=24 18e021bcace225347459189f292fbc2e4677563b07498a07
ah=md5 key=16 b4f44368741632b4e33e5f5b794253d3
enc: spi=ae01010d esp=3des key=24 42c94a8a2f72a44f9a3777f8e6aa3b24160b8af15f54a573
ah=md5 key=16 6214155f76b63a93345dcc9ec02d6415
dec:pkts/bytes=3073/4299621477, enc:pkts/bytes=1667/66375
 npu_flag=03 npu_rgwy=11.11.11.2 npu_lgwy=11.11.11.1 npu_selid=4

diagnose sys session list
session info: proto=6 proto_state=01 duration=34 expire=3565 timeout=3600
    flags=00000000 sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
```

```
ha_id=0 policy_dir=0 tunnel=/p1-vdom2
state=re may_dirty npu
statistic(bytes/packets/allow_err): org=112/2/1 reply=112/2/1 tuples=2
orgin->sink: org pre->post, reply pre->post dev=57->7/7->57 gwy=10.1.100.11/11.11.11.1
hook=pre dir=org act=noop 172.16.200.55:35254->10.1.100.11:80(0.0.0.0:0)
hook=post dir=reply act=noop 10.1.100.11:80->172.16.200.55:35254(0.0.0.0:0)
pos/(before,after) 0/(0,0), 0/(0,0)
misc=0 policy_id=1 id_policy_id=0 auth_info=0 chk_client_info=0 vd=4
serial=00002d29 tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=172.16.200.55, bps=260
npu_state=00000000
npu info: flag=0x81/0x82, offload=7/7, ips_offload=0/0, epid=1/3, ipid=3/1,
    vlan=32779/0
```

Controlling IPS NPx and CPx acceleration

You can use the following commands to enale or disable acceleration of IPS processing by NPx and CPx processors:

```
config ips global
   set np-accel-mode {none | basic}
   set cp-accel-mode {none | basic | advanced}
end
```

The network processor (NP) acceleration modes are:

`none`: Network Processor acceleration disabled

`basic`: Basic Network Processor acceleration enabled

The content processor (CP) acceleration modes are:

`none`: Content Processor acceleration disabled

`basic`: Basic Content Processor acceleration enabled

`advanced`: Advanced Content Processor acceleration enabled

Dedicated Management CPU

The web-based manager and CLI of FortiGate units with NP6 and NP4 processors may become unresponsive when the system is under heavy processing load because NP6 or NP4 interrupts overload the CPUs preventing CPU cycles from being used for management tasks. You can resolve this issue by using the following command to dedicate CPU core 0 to management tasks.

```
config system npu
   set dedicated-management-cpu {enable | disable}
end
```

All management tasks are then processed by CPU 0 and NP6 or NP4 interrupts are handled by the remaining CPU cores.

NP6 Acceleration

NP6 network processors provide fastpath acceleration by offloading communication sessions from the FortiGate CPU. When the first packet of a new session is received by an interface connected to an NP6 processor, just like any session connecting with any FortiGate interface, the session is forwarded to the FortiGate CPU where it is matched with a security policy. If the session is accepted by a security policy and if the session can be offloaded its session key is copied to the NP6 processor that received the packet. All of the rest of the packets in the session are intercepted by the NP6 processor and fast-pathed out of the FortiGate unit to their destination without ever passing through the FortiGate CPU. The result is enhanced network performance provided by the NP6 processor plus the network processing load is removed from the CPU. In addition the NP6 processor can handle some CPU intensive tasks, like IPsec VPN encryption/decryption.

Session keys (and IPsec SA keys) are stored in the memory of the NP6 processor that is connected to the interface that received the packet that started the session. All sessions are fast-pathed and accelerated, even if they exit the FortiGate unit through an interface connected to another NP6. There is no dependence on getting the right pair of interfaces since the offloading is done by the receiving NP6.

The key to making this possible is an Integrated Switch Fabric (ISF) that connects the NP6s and the FortiGate unit interfaces together. Many FortiGate units with NP6 processors also have an ISF. The ISF allows any port connectivity. All ports and NP6s can communicate with each other over the ISF. There are no special ingress and egress fast path requirements as long as traffic enters and exits on interfaces connected to the same ISF.

Some FortiGate units, such as the FortiGate-1000D include multiple NP6 processors that are not connected by an ISF. Because the ISF is not present fast path acceleration is supported only between interfaces connected to the same NP6 processor. Since the ISF introduces some latency, models with no ISF provide low-latency network acceleration between network interfaces connected to the same NP6 processor.

There are at least two limitations to keep in mind:

- The capacity of each NP6 processor. An individual NP6 processor can support between 10 and 16 million sessions. This number is limited by the amount of memory the processor has. Once an NP6 processor hits its session limit, sessions that are over the limit are sent to the CPU. You can avoid this problem by as much as possible distributing incoming sessions evenly among the NP6 processors. To be able to do this you need to be aware of which interfaces connect to which NP6 processors and distribute incoming traffic accordingly.
- Some FortiGate units may use some NP6 processors for special functions. For example, ports 25 to 32 of the FortiGate-3700D can be used for low latency offloading.

NP6 session fast path requirements

NP6 processors can offload the following traffic and services:

- IPv4 and IPv6 traffic and NAT64 and NAT46 traffic (as well as IPv4 and IPv6 versions of the following traffic types where appropriate)
- TCP, UDP, ICMP and SCTP traffic
- IPSec VPN traffic, and offloading of IPsec encryption/decryption (including SHA2-256 and SHA2-512)
- Anomaly-based intrusion prevention, checksum offload and packet defragmentation
- SIT and IPv6 Tunnelling sessions

- Multicast traffic (including Multicast over IPsec)
- CAPWAP and wireless bridge traffic tunnel encapsulation to enable line rate wireless forwarding from FortiAP devices
- Traffic shaping and priority queuing for both shared and per IP traffic shaping. An NP6 processor has 16 million queues for traffic shaping and statistics counting.
- Syn proxying
- Inter-VDOM link traffic

Sessions that are offloaded must be fast path ready. For a session to be fast path ready it must meet the following criteria:

- Layer 2 type/length must be 0x0800 for IPv4 or 0x86dd for IPv6 (IEEE 802.1q VLAN specification is supported)
- Link aggregation between any network interfaces sharing the same network processor(s) may be used (IEEE 802.3ad specification is supported)
- Layer 3 protocol can be IPv4 or IPv6
- Layer 4 protocol can be UDP, TCP, ICMP, or SCTP
- In most cases, Layer 3 / Layer 4 header or content modification sessions that require a session helper can be offloaded.
- Local host traffic (originated by the FortiGate unit) can be offloaded
- Application layer content modification is not supported (the firewall policy that accepts the session must not include virus scanning, web filtering, DLP, application control, IPS, email filtering, SSL/SSH inspection, VoIP or ICAP)

If you disable anomaly checks by Intrusion Prevention (IPS), you can still enable hardware accelerated anomaly checks using the `fp-anomaly` field of the `config system interface` CLI command. See Configuring individual NP6 processors on page 26.

If a session or is not fast path ready, the FortiGate unit will not send the session key or IPsec SA key to the NP6 processor. Without the session key, all session key lookup by a network processor for incoming packets of that session fails, causing all session packets to be sent to the FortiGate unit's main processing resources, and processed at normal speeds.

If a session is fast path ready, the FortiGate unit will send the session key or IPsec SA key to the network processor. Session key or IPsec SA key lookups then succeed for subsequent packets from the known session or IPsec SA.

Packet fast path requirements

Packets within the session must then also meet packet requirements.

- Incoming packets must not be fragmented.
- Outgoing packets must not require fragmentation to a size less than 385 bytes. Because of this requirement, the configured MTU (Maximum Transmission Unit) for a network processor's network interfaces must also meet or exceed the network processors' supported minimum MTU of 385 bytes.

Mixing fast path and non-fast path traffic

If packet requirements are not met, an individual packet will be processed by the FortiGate CPU regardless of whether other packets in the session are offloaded to the NP6.

Also, in some cases, a protocol's session(s) may receive a mixture of offloaded and non-offloaded processing. For example, VoIP control packets may not be offloaded but VoIP data packets (voice packets) may be offloaded.

Viewing your FortiGate NP6 processor configuration

Use the following command to view the NP6 processor configuration of your FortiGate unit:

```
diagnose npu np6 port-list
```

For example output of this command for different FortiGate models, see FortiGate NP6 architectures on page 32.

Increasing NP6 offloading capacity using link aggregation groups (LAGs)

NP6 processors can offload sessions received by interfaces in link aggregation groups (LAGs). A LAG combines more than one physical interface into a group that functions like a single interface with a higher capacity than a single physical interface. For example, you could use a LAG if you want to offload sessions on a 30 GB link by adding three 10-GB interfaces to the same LAG.

Just like with normal interfaces, traffic accepted by a LAG is offloaded by the NP6 processor connected to the interfaces in the LAG that receive the traffic to be offloaded. If all interfaces in a LAG are connected to the same NP6 processor, traffic received by that LAG is offloaded by that NP6 processor. The amount of traffic that can be offloaded is limited by the capacity of the NP6 processor.

If a FortiGate has two or more NP6 processors, you can use LAGs to increase offloading by sharing the traffic load across multiple NP6 processors. You do this by adding physical interfaces connected to different NP6 processors to the same LAG.

Adding a second NP6 processor to a LAG effectively doubles the offloading capacity of the LAG. Adding a third further increases offloading. The actual increase in offloading capacity may not actually be doubled by adding a second NP6 or tripled by adding a thrid. Traffic and load conditions and other factors may limit the actual offloading result.

The increase in offloading capacity offered by LAGs and multiple NP6s is supported by the integrated switch fabric (ISF) that allows multiple NP6 processors to share session information. Most FortiGate units with multiple NP6 processors also have an ISF. However, the FortiGate-1000D does not have an ISF. On this model and others that have more than one NP6 and no ISF, if you attempt to add interfaces connected to different NP6 processors to a LAG the system displays an error message.

Configuring Inter-VDOM link acceleration with NP6 processors

FortiGate units with NP6 processors include inter-VDOM links that can be used to accelerate inter-VDOM link traffic.

- For a FortiGate unit with two NP6 processors there are two accelerated inter-VDOM links, each with two interfaces:
  - **npu0_vlink:**
    npu0_vlink0
    npu0_vlink1

- **npu1_vlink:**
  npu1_vlink0
  npu1_vlink1

These interfaces are visible from the GUI and CLI. For a FortiGate unit with NP6 interfaces, enter the following CLI command to display the NP6-accelerated inter-VDOM links:

```
get system interface
...
== [ npu0_vlink0 ]
name: npu0_vlink0 mode: static ip: 0.0.0.0 0.0.0.0 status: down netbios-forward: disable
    type: physical sflow-sampler: disable explicit-web-proxy: disable explicit-ftp-proxy:
    disable mtu-override: disable wccp: disable drop-overlapped-fragment: disable drop-
    fragment: disable
== [ npu0_vlink1 ]
name: npu0_vlink1 mode: static ip: 0.0.0.0 0.0.0.0 status: down netbios-forward: disable
    type: physical sflow-sampler: disable explicit-web-proxy: disable explicit-ftp-proxy:
    disable mtu-override: disable wccp: disable drop-overlapped-fragment: disable drop-
    fragment: disable
== [ npu1_vlink0 ]
name: npu1_vlink0 mode: static ip: 0.0.0.0 0.0.0.0 status: down netbios-forward: disable
    type: physical sflow-sampler: disable explicit-web-proxy: disable explicit-ftp-proxy:
    disable mtu-override: disable wccp: disable drop-overlapped-fragment: disable drop-
    fragment: disable
== [ npu1_vlink1 ]
name: npu1_vlink1 mode: static ip: 0.0.0.0 0.0.0.0 status: down netbios-forward: disable
    type: physical sflow-sampler: disable explicit-web-proxy: disable explicit-ftp-proxy:
    disable mtu-override: disable wccp: disable drop-overlapped-fragment: disable drop-
    fragment: disable
...
```

By default the interfaces in each inter-VDOM link are assigned to the root VDOM. To use these interfaces to accelerate inter-VDOM link traffic, assign each interface in the pair to the VDOMs that you want to offload traffic between. For example, if you have added a VDOM named New-VDOM to a FortiGate unit with NP4 processors, you can go to **System > Network > Interfaces** and edit the **npu0-vlink1** interface and set the **Virtual Domain** to **New-VDOM**. This results in an accelerated inter-VDOM link between root and New-VDOM. You can also do this from the CLI:

```
config system interface
   edit npu0-vlink1
      set vdom New-VDOM
end
```

Using VLANs to add more accelerated Inter-VDOM links

You can add VLAN interfaces to the accelerated inter-VDOM links to create inter-VDOM links between more VDOMs. For the links to work, the VLAN interfaces must be added to the same inter-VDOM link, must be on the same subnet, and must have the same VLAN ID.

For example, to accelerate inter-VDOM link traffic between VDOMs named Marketing and Engineering using VLANs with VLAN ID 100 go to **System > Network > Interfaces** and select **Create New** to create the VLAN interface associated with the Marketing VDOM:

Name	Marketing-link

Type	VLAN
Interface	npu0_vlink0
VLAN ID	100
Virtual Domain	Marketing
IP/Network Mask	172.20.120.12/24

Create the inter-VDOM link associated with Engineering VDOM:

Name	Engineering-link
Type	VLAN
Interface	npu0_vlink1
VLAN ID	100
Virtual Domain	Engineering
IP/Network Mask	172.20.120.22/24

Or do the same from the CLI:

```
config system interface
   edit Marketing-link
      set vdom Marketing
      set ip 172.20.120.12/24
      set interface npu0_vlink0
      set vlanid 100
   next
   edit Engineering-link
      set vdom Engineering
      set ip 172.20.120.22/24
      set interface npu0_vlink1
      set vlanid 100
```

Confirm that the traffic is accelerated

Use the following CLI commands to obtain the interface index and then correlate them with the session entries. In the following example traffic was flowing between new accelerated inter-VDOM links and physical ports port1 and port 2 also attached to the NP6 processor.

```
diagnose ip address list
IP=172.31.17.76->172.31.17.76/255.255.252.0 index=5 devname=port1
IP=10.74.1.76->10.74.1.76/255.255.252.0 index=6 devname=port2
IP=172.20.120.12->172.20.120.12/255.255.255.0 index=55 devname=IVL-VLAN1_ROOT
IP=172.20.120.22->172.20.120.22/255.255.255.0 index=56 devname=IVL-VLAN1_VDOM1

diagnose sys session list
```

```
session info: proto=1 proto_state=00 duration=282 expire=24 timeout=0 session info:
    proto=1 proto_state=00 duration=124 expire=59 timeout=0 flags=00000000
    sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=180/3/1 reply=120/2/1 tuples=2
orgin->sink: org pre->post, reply pre->post dev=55->5/5->55
    gwy=172.31.19.254/172.20.120.22
hook=post dir=org act=snat 10.74.2.87:768->10.2.2.2:8(172.31.17.76:62464)
hook=pre dir=reply act=dnat 10.2.2.2:62464->172.31.17.76:0(10.74.2.87:768)
misc=0 policy_id=4 id_policy_id=0 auth_info=0 chk_client_info=0 vd=0
serial=0000004e tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=10.74.2.87, bps=880
npu_state=00000000
npu info: flag=0x81/0x81, offload=8/8, ips_offload=0/0, epid=160/218, ipid=218/160,
    vlan=32769/0


session info: proto=1 proto_state=00 duration=124 expire=20 timeout=0 flags=00000000
    sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=180/3/1 reply=120/2/1 tuples=2
orgin->sink: org pre->post, reply pre->post dev=6->56/56->6
    gwy=172.20.120.12/10.74.2.87
hook=pre dir=org act=noop 10.74.2.87:768->10.2.2.2:8(0.0.0.0:0)
hook=post dir=reply act=noop 10.2.2.2:768->10.74.2.87:0(0.0.0.0:0)
misc=0 policy_id=3 id_policy_id=0 auth_info=0 chk_client_info=0 vd=1
serial=0000004d tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=10.74.2.87, bps=880
npu_state=00000000
npu info: flag=0x81/0x81, offload=8/8, ips_offload=0/0, epid=219/161, ipid=161/219,
    vlan=0/32769
total session 2
```

Configuring individual NP6 processors

You can use the following command to configure a wide range of settings for the NP6 processors in your FortiGate unit including enabling/disabling fastpath and low latency and adjusting session timeouts. As well you can set anomaly checking for IPv4 and IPv6 traffic. You can also configure different settings for each NP6 processor.

Some of the options for this command apply anomaly checking for NP6 sessions in the same was as the command descrbed in Offloading NP4 anomaly detection on page 60 applies anomaly checking for for NP4 sessions.

```
config system np6
   edit <np6-processor-name>
      set fastpath {disable | enable}
      set low-latency-mode {disable | enable}
      set session-timeout-random-range <range>
      set garbage-session-collector {disable | enable}
      set session-collector-interval <range>
      set session-timeout-interval <range>
      set session-timeout-random-range <range>
      set session-timeout-fixed {disable | enable}
         config fp-anomaly-v4
            set icmp-frag {allow | drop | trap-to-host}
            set icmp-land {allow | drop | trap-to-host}
            set ipv4-land {allow | drop | trap-to-host}
            set ipv4-optlsrr {allow | drop | trap-to-host}
            set ipv4-optrr {allow | drop | trap-to-host}
            set ipv4-optsecurity {allow | drop | trap-to-host}
            set ipv4-optssrr {allow | drop | trap-to-host}
            set ipv4-optstream {allow | drop | trap-to-host}
            set ipv4-opttimestamp {allow | drop | trap-to-host}
            set ipv4-proto-err {allow | drop | trap-to-host}
            set ipv4-unknopt {allow | drop | trap-to-host}
            set tcp-land {allow | drop | trap-to-host}
            set tcp-syn-fin {allow | drop | trap-to-host}
            set tcp-winnuke {allow | drop | trap-to-host}
            set tcp_fin_noack {allow | drop | trap-to-host}
            set tcp_fin_only {allow | drop | trap-to-host}
            set tcp_no_flag {allow | drop | trap-to-host}
            set tcp_syn_data {allow | drop | trap-to-host}
            set udp-land {allow | drop | trap-to-host}
         end
         config fp-anomaly-v6
            set ipv6-daddr_err {allow | drop | trap-to-host}
            set ipv6-land {allow | drop | trap-to-host}
            set ipv6-optendpid {allow | drop | trap-to-host}
            set ipv6-opthomeaddr {allow | drop | trap-to-host}
            set ipv6-optinvld {allow | drop | trap-to-host}
            set ipv6-optjumbo {allow | drop | trap-to-host}
            set ipv6-optnsap {allow | drop | trap-to-host}
            set ipv6-optralert {allow | drop | trap-to-host}
            set ipv6-opttunnel {allow | drop | trap-to-host}
            set ipv6-proto-err {allow | drop | trap-to-host}
            set ipv6-saddr_err {allow | drop | trap-to-host}
            set ipv6-unknopt {allow | drop | trap-to-host}
         end
```

Command syntax

Command	Description	Default
`fastpath {disable | enable}`	Enable fastpath acceleration to offload sessions to the NP6 processor. You can disable fastpath if you don't want the NP6 processor to offload sessions.	enable

Command	Description	Default
`low-latency-mode {disable | enable}`	Enable low-latency mode. In low latency mode the integrated switch fabric is bypassed. Low latency mode requires that packet enter and exit using the same NP6 processor. This option is only available for NP6 processors that can operate in low-latency mode.	disable
`per-session-accounting {disable | enable}`	Record traffic log messages for offloaded sessions. Enabling this feature reduces performance. See “Enabling per-session accounting for offloaded NP6 sessions” on page 32.	disable
`garbage-session-collector {disable | enable}`	Enable deleting expired or garbage sessions.	disable
`session-collector-interval <range>`	Set the expired or garbage session collector time interval in seconds. The range is 1 to 100 seconds.	8
`session-timeout-interval <range>`	Set the timeout for inactive sessions. The range is 0 to 1000 seconds.	40
`session-timeout-random-range <range>`	Set the random timeout for inactive sessions. The range is 0 to 1000 seconds.	8
`session-timeout-fixed {disable | enable}`	Force session timeouts at fixed instead of random intervals.	disable
config fp-anomaly-v4		
`fp-anomaly-v4`	Configure how the NP6 processor does IPv4 traffic anomaly protection. You can configure the NP6 processor to allow or drop the packets associated with an attack or forward the packets that are associated with the attack to FortiOS (called “trap-to-host”). Selecting “trap-to-host” turns off NP6 anomaly protection for that anomaly. If you require anomaly protection you can enable it with a DoS policy.	
`icmp-frag {allow | drop | trap-to-host}`	Detects Layer 3 fragmented packets that could be part of a layer 4 ICMP anomalies.	allow
`icmp-land {allow | drop | trap-to-host}`	Detects ICMP land anomalies.	trap-to-host
`ipv4-land {allow | drop | trap-to-host}`	Detects IPv4 land anomalies.	trap-to-host

Command	Description	Default
`ipv4-optlsrr {allow | drop | trap-to-host}`	Detects IPv4 with loose source record route option anomalies.	trap-to-host
`ipv4-optrr {allow | drop | trap-to-host}`	Detects IPv4 with record route option anomalies.	trap-to-host
`ipv4-optsecurity {allow | drop | trap-to-host}`	Detects security option anomalies.	trap-to-host
`ipv4-optssrr {allow | drop | trap-to-host}`	Detects IPv4 with strict source record route option anomalies.	trap-to-host
`ipv4-optstream {allow | drop | trap-to-host}`	Detects stream option anomalies.	trap-to-host
`ipv4-opttimestamp {allow | drop | trap-to-host}`	Detects timestamp option anomalies.	trap-to-host
`ipv4-proto-err {allow | drop | trap-to-host}`	Detects invalid layer 4 protocol anomalies.	trap-to-host
`ipv4-unknopt {allow | drop | trap-to-host}`	Detects unknown option anomalies.	trap-to-host
`tcp-land {allow | drop | trap-to-host}`	Detects TCP land anomalies.	trap-to-host
`tcp-syn-fin {allow | drop | trap-to-host}`	Detects TCP SYN flood SYN/FIN flag set anomalies.	allow
`tcp-winnuke {allow | drop | trap-to-host}`	Detects TCP WinNuke anomalies.	trap-to-host
`tcp_fin_noack {allow | drop | trap-to-host}`	Detects TCP SYN flood with FIN flag set without ACK setting anomalies.	trap-to-host
`tcp_fin_only {allow | drop | trap-to-host}`	Detects TCP SYN flood with only FIN flag set anomalies.	trap-to-host
`tcp_no_flag {allow | drop | trap-to-host}`	Detects TCP SYN flood with no flag set anomalies.	allow
`tcp_syn_data {allow | drop | trap-to-host}`	Detects TCP SYN flood packets with data anomalies.	allow
`udp-land {allow | drop | trap-to-host}`	Detects UDP land anomalies.	trap-to-host

Command	Description	Default
config fp-anomaly-v6		
`fp-anomaly-v6`	Configure how the NP6 processor does IPv6 traffic anomaly protection. You can configure the NP6 processor to allow or drop the packets associated with an attack or forward the packets that are associated with the attack to FortiOS (called “trap-to-host”). Selecting “trap-to-host” turns off NP6 anomaly protection for that anomaly. If you require anomaly protection you can enable it with a DoS policy.	
`ipv6-daddr_err {allow | drop | trap-to-host}`	Detects destination address as unspecified or loopback address anomalies.	trap-to-host
`ipv6-land {allow | drop | trap-to-host}`	Detects IPv6 land anomalies	trap-to-host
`ipv6-optendpid {allow | drop | trap-to-host}`	Detects end point identification anomalies.	trap-to-host
`ipv6-opthomeaddr {allow | drop | trap-to-host}`	Detects home address option anomalies.	trap-to-host
`ipv6-optinvld {allow | drop | trap-to-host}`	Detects invalid option anomalies.	trap-to-host
`ipv6-optjumbo {allow | drop | trap-to-host}`	Detects jumbo options anomalies.	trap-to-host
`ipv6-optnsap {allow | drop | trap-to-host}`	Detects network service access point address option anomalies.	trap-to-host
`ipv6-optralert {allow | drop | trap-to-host}`	Detects router alert option anomalies.	trap-to-host
`ipv6-opttunnel {allow | drop | trap-to-host}`	Detects tunnel encapsulation limit option anomalies.	trap-to-host
`ipv6-proto-err {allow | drop | trap-to-host}`	Detects layer 4 invalid protocol anomalies.	trap-to-host
`ipv6-saddr_err {allow | drop | trap-to-host}`	Detects source address as multicast anomalies.	trap-to-host
`ipv6-unknopt {allow | drop | trap-to-host}`	Detects unknown option anomalies.	trap-to-host

Enabling per-session accounting for offloaded NP6 sessions

By default FortiOS does not record log messages for offloaded NP6 and NP4 sessions. This also means that traffic monitoring does not report correct session counts, byte counts and packet counts for offloaded NP6 and NP4 sessions.

However, for NP6 processors, you can use the following command to enable per-session accounting for each NP6 processor in the FortiGate unit. Per session accounting is a logging feature that allows the FortiGate to report the correct bytes/pkt numbers per session for sessions offloaded to an NP6 processor. Per-session accounting is not enabled by default because it reduces NP6 offloading performance. So you should only enable per-session accounting if you need the accounting information.

For example, to enable session accounting for the first and second NP6 processors (np6_0 and np6_1):

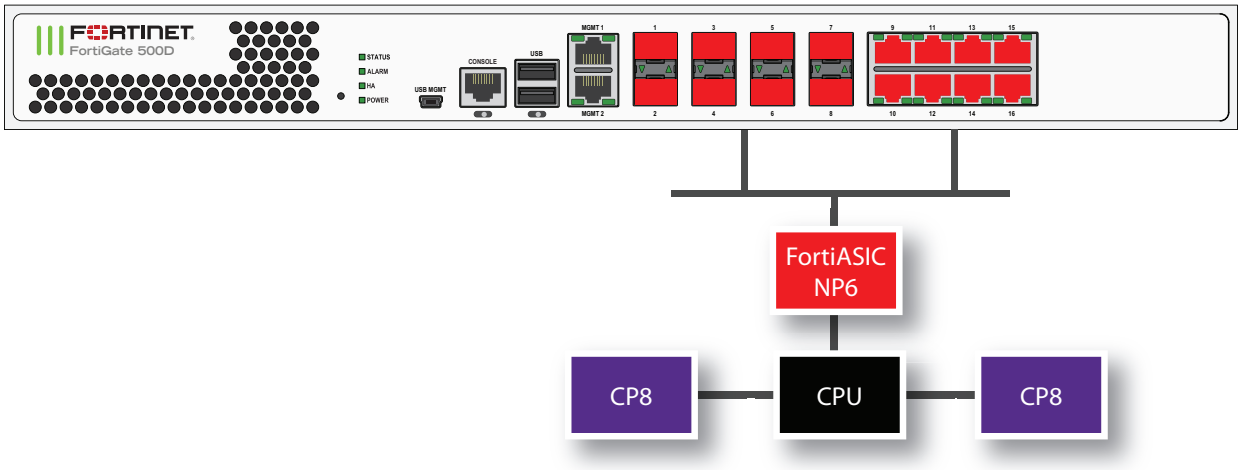
```
config system np6
   edit np6_0
      set per-session-acounting enable
   next
   edit np6_1
      set per-session-acounting enable
   end
```

FortiGate NP6 architectures

Many FortiGate models can offload some types of network traffic processing from main processing resources to specialized network processors. If your network has a significant volume of traffic that is suitable for offloading, this hardware acceleration can significantly improve your network throughput.

FortiGate-500D fast path architecture

The FortiGate-500D includes one NP6 processor connected to eight 1Gb SFP interfaces (port1-port8 and eight 1Gb RJ-45 Ethernet ports (port9-16).



You can use the following command to display the FortiGate-500D NP6 configuration. The command output shows one NP6 named NP6_0 and the interfaces (ports) connected to it.

```
diagnose npu np6 port-list
Chip   XAUI Ports   Max   Cross-chip
                    Speed offloading
------ ---- ------- ----- ----------
np6_0  0
       1    port10  1G    Yes
       1    port9   1G    Yes
       1    port12  1G    Yes
       1    port11  1G    Yes
       1    port14  1G    Yes
       1    port13  1G    Yes
       1    port16  1G    Yes
       1    port15  1G    Yes
       1    port5   1G    Yes
       1    port7   1G    Yes
       1    port8   1G    Yes
       1    port6   1G    Yes
       1    port3   1G    Yes
       1    port4   1G    Yes
       1    port1   1G    Yes
```

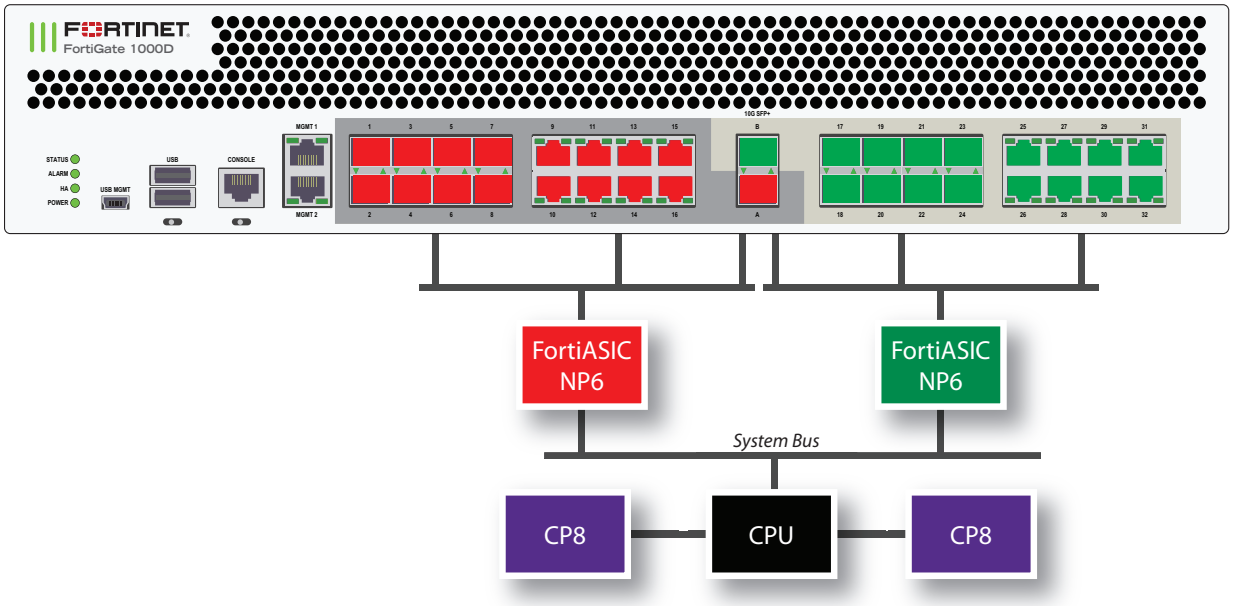
```
          1    port2   1G    Yes
          2
          3
------ ---- ------- ----- ----------
```

FortiGate-1000D fast path architecture

The FortiGate-1000D includes two NP6 processors that are not connected by an integrated switch fabric (ISF). The NP6 processors are connected to network interfaces as follows:

Because the FortiGate-1000D does not have an ISF you cannot create Link Aggregation Groups (LAGs) that include interfaces connected to both NP6 processors.

- Eight 1Gb SFP interfaces (port17-port24), eight 1Gb RJ-45 Ethernet interfaces (port25-32) and one 10Gb SFP+ interface (portB) share connections to the first NP6 processor.
- Eight 1Gb SFP interfaces (port1-port8), eight RJ-45 Ethernet interfaces (port9-16) and one 10Gb SFP+ interface (portA) share connections to the second NP6 processor.



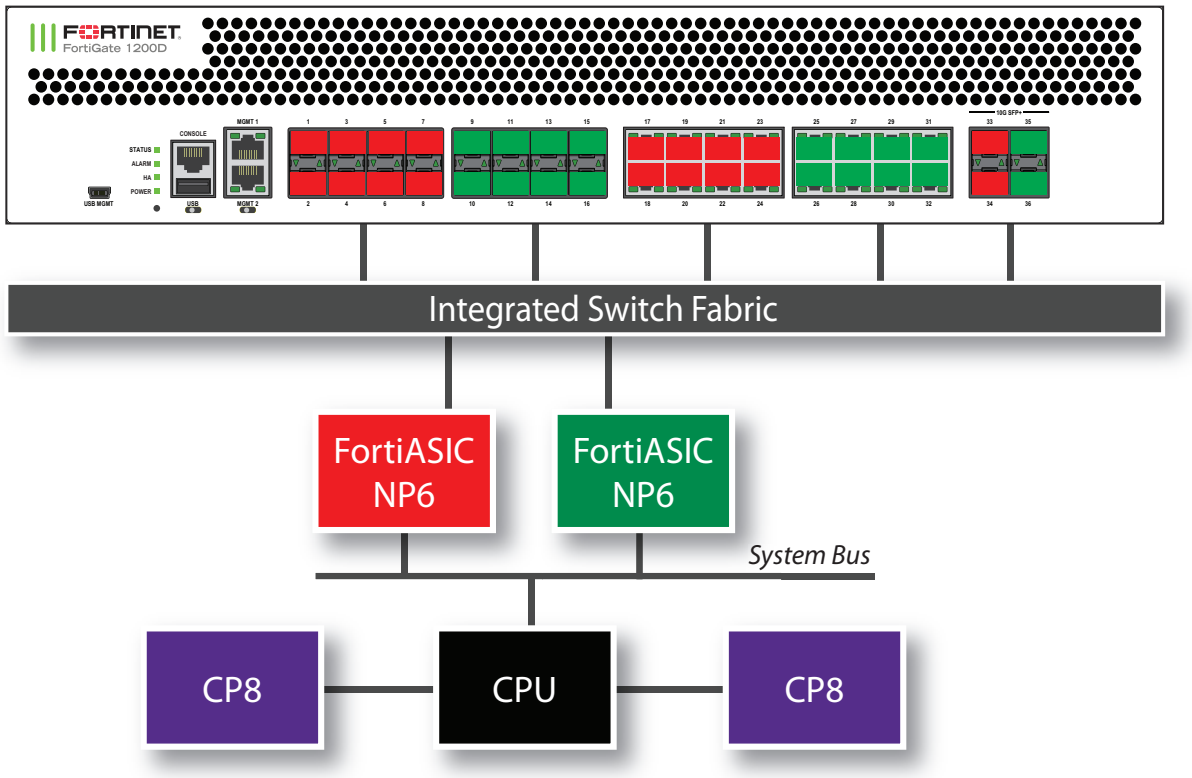
You can use the following command to display the FortiGate-1000D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6.

```
diagnose npu np6 port-list
Chip   XAUI Ports   Max   Cross-chip
                    Speed offloading
------ ---- ------- ----- ----------
np6_0  0
       1    port17  1G    Yes
       1    port18  1G    Yes
       1    port19  1G    Yes
       1    port20  1G    Yes
       1    port21  1G    Yes
       1    port22  1G    Yes
       1    port23  1G    Yes
       1    port24  1G    Yes
       1    port27  1G    Yes
       1    port28  1G    Yes
       1    port25  1G    Yes
       1    port26  1G    Yes
       1    port31  1G    Yes
       1    port32  1G    Yes
       1    port29  1G    Yes
       1    port30  1G    Yes
       2    portB   10G   Yes
       3
------ ---- ------- ----- ----------
np6_1   0
        1    port1  1G    Yes
        1    port2  1G    Yes
        1    port3  1G    Yes
        1    port4  1G    Yes
        1    port5  1G    Yes
        1    port6  1G    Yes
        1    port7  1G    Yes
        1    port8  1G    Yes
        1    port11 1G    Yes
        1    port12 1G    Yes
        1    port9  1G    Yes
        1    port10 1G    Yes
        1    port15 1G    Yes
        1    port16 1G    Yes
        1    port13 1G    Yes
        1    port14 1G    Yes
        2    portA  10G   Yes
        3
```

FortiGate-1200D fast path architecture

The FortiGate-1200D features two NP6 processors both connected to an integrated switch fabric.

- Eight SFP 1Gb interfaces (port1-port8), eight RJ-45 Ethernet ports (port17-24) and two SFP+ 10Gb interfaces (port33 and port34) share connections to the first NP6 processor.
- Eight SFP 1Gb interfaces (port9-port16), eight RJ-45 Ethernet ports (port25-32) and two SFP+ 10Gb interfaces (port35-port36) share connections to the second NP6 processor.



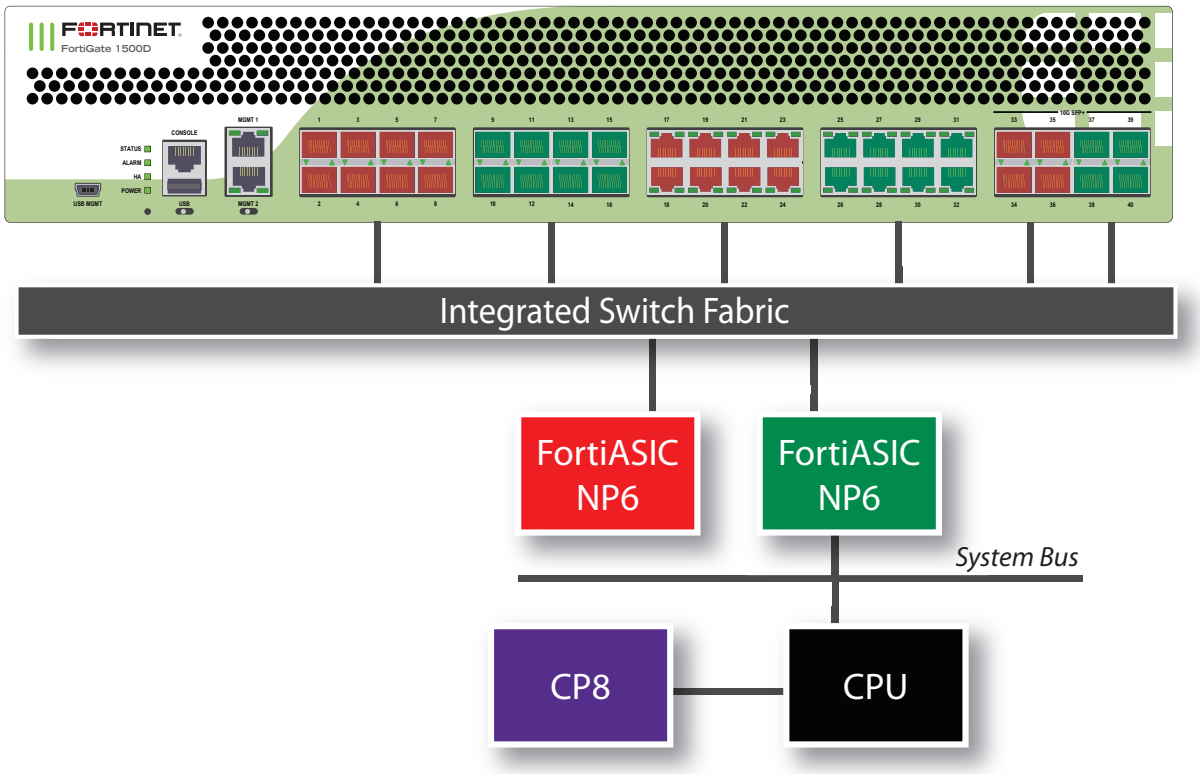
You can use the following command to display the FortiGate-1200D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6.

```
diag npu np6 port-list
Chip   XAUI Ports   Max   Cross-chip
                    Speed offloading
------ ---- ------- ----- ----------
np6_0  0    port33  10G   Yes
       1    port34  10G   Yes
       2    port1   1G    Yes
       2    port3   1G    Yes
       2    port5   1G    Yes
       2    port7   1G    Yes
       2    port17  1G    Yes
       2    port19  1G    Yes
       2    port21  1G    Yes
       2    port23  1G    Yes
       3    port2   1G    Yes
       3    port4   1G    Yes
       3    port6   1G    Yes
       3    port8   1G    Yes
       3    port18  1G    Yes
       3    port20  1G    Yes
       3    port22  1G    Yes
       3    port24  1G    Yes
------ ---- ------- ----- ----------
np6_1  0    port35  10G   Yes
       1    port36  10G   Yes
       2    port9   1G    Yes
       2    port11  1G    Yes
       2    port13  1G    Yes
       2    port15  1G    Yes
       2    port25  1G    Yes
       2    port27  1G    Yes
       2    port29  1G    Yes
       2    port31  1G    Yes
       3    port10  1G    Yes
       3    port12  1G    Yes
       3    port14  1G    Yes
       3    port16  1G    Yes
       3    port26  1G    Yes
       3    port28  1G    Yes
       3    port30  1G    Yes
       3    port32  1G    Yes
------ ---- ------- ----- ----------
```

FortiGate-1500D fast path architecture

The FortiGate-1500D features two NP6 processors both connected to an integrated switch fabric.

- Eight SFP 1Gb interfaces (port1-port8), eight RJ-45 Ethernet ports (port17-24) and four SFP+ 10Gb interfaces (port33-port36) share connections to the first NP6 processor.
- Eight SFP 1Gb interfaces (port9-port16), eight RJ-45 Ethernet ports (port25-32) and four SFP+ 10Gb interfaces (port37-port40) share connections to the second NP6 processor.



You can use the following command to display the FortiGate-1500D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6.

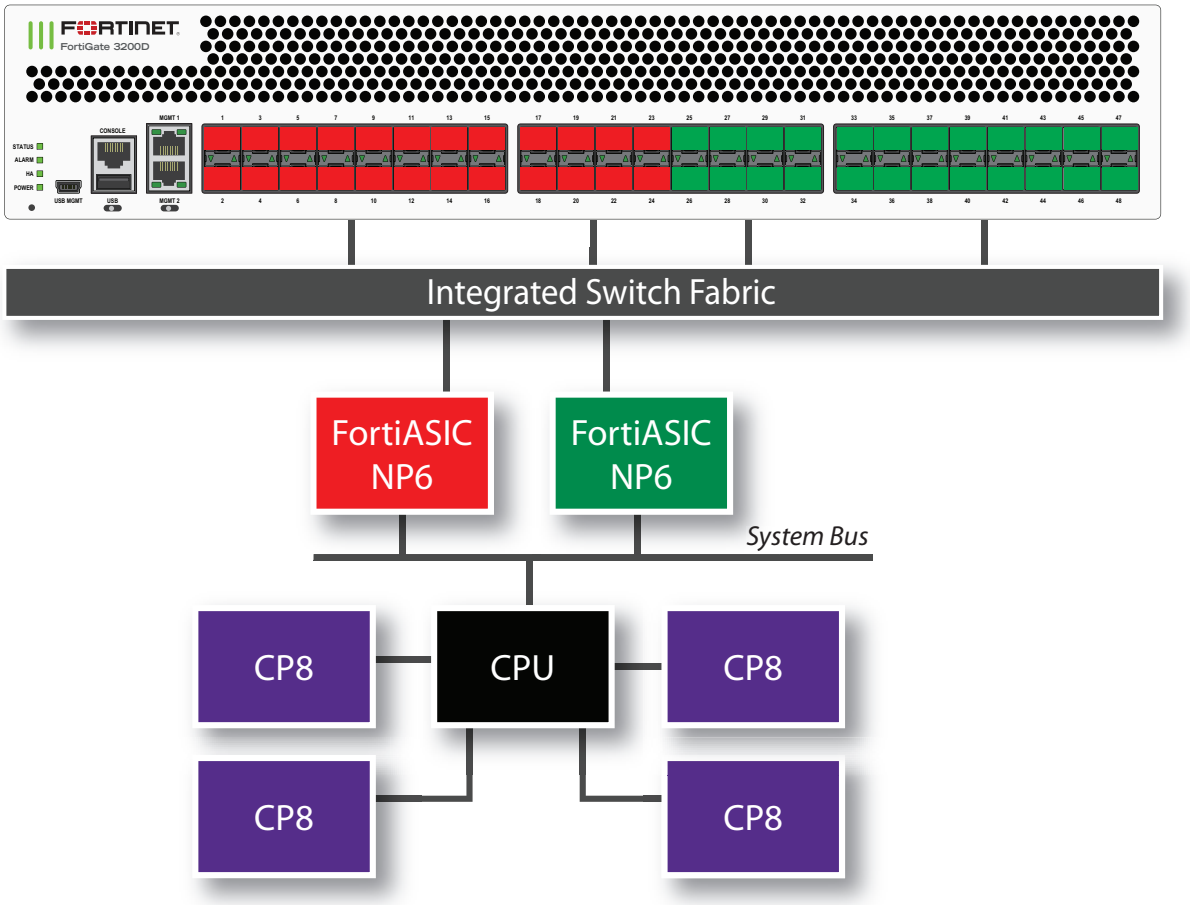
```
diagnose npu np6 port-list
Chip  XAUI Ports Max Cross-chip
  Speed offloading
------ ---- ------- ----- ----------
np6_0   0   port1   10G   Yes
         0   port5   10G   Yes
         0   port17  10G   Yes
         0   port21  10G   Yes
         0   port33  10G   Yes
         1   port2   10G   Yes
         1   port6   10G   Yes
         1   port18  10G   Yes
         1   port22  10G   Yes
         1   port34  10G   Yes
         2   port3   10G   Yes
```

```
          2   port7   10G   Yes
          2   port19  10G   Yes
          2   port23  10G   Yes
          2   port35  10G   Yes
          3   port4   10G   Yes
          3   port8   10G   Yes
          3   port20  10G   Yes
          3   port24  10G   Yes
          3   port36  10G   Yes
------ ---- ------- ----- ----------
np6_1   0   port9   10G   Yes
          0   port13  10G   Yes
          0   port25  10G   Yes
          0   port29  10G   Yes
          0   port37  10G   Yes
          1   port10  10G   Yes
          1   port14  10G   Yes
          1   port26  10G   Yes
          1   port30  10G   Yes
          1   port38  10G   Yes
          2   port11  10G   Yes
          2   port15  10G   Yes
          2   port27  10G   Yes
          2   port31  10G   Yes
          2   port39  10G   Yes
          3   port12  10G   Yes
          3   port16  10G   Yes
          3   port28  10G   Yes
          3   port32  10G   Yes
          3   port40  10G   Yes
------ ---- ------- ----- ----------
```

FortiGate-3200D fast path architecture

The FortiGate-3200D features two NP6 processors connected to an Integrated Switch Fabirc (ISF). The FortiGate-3200D has the following fastpath architecture:

- 24 SFP+ 10Gb interfaces, port1 through port24 share connections to the first NP6 processor (np6_0).
- 24 SFP+ 10Gb interfaces, port25 through port48 share connections to the first NP6 processor (np6_1).



You can use the following command to display the FortiGate-3200D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1 and the interfaces (ports) connected to each NP6.

```
diag npu np6 port-list
Chip    XAUI Ports    Max    Cross-chip
                      Speed  offloading
------  ---- -------  -----  ----------
np6_0   0    port1    10G    Yes
        0    port5    10G    Yes
        0    port10   10G    Yes
        0    port13   10G    Yes
        0    port17   10G    Yes
        0    port22   10G    Yes
```

```
        1    port2   10G   Yes
        1    port6   10G   Yes
        1    port9   10G   Yes
        1    port14  10G   Yes
        1    port18  10G   Yes
        1    port21  10G   Yes
        2    port3   10G   Yes
        2    port7   10G   Yes
        2    port12  10G   Yes
        2    port15  10G   Yes
        2    port19  10G   Yes
        2    port24  10G   Yes
        3    port4   10G   Yes
        3    port8   10G   Yes
        3    port11  10G   Yes
        3    port16  10G   Yes
        3    port20  10G   Yes
        3    port23  10G   Yes
------ ---- ------- ----- ----------
np6_1  0    port26  10G   Yes
        0    port29  10G   Yes
        0    port33  10G   Yes
        0    port37  10G   Yes
        0    port41  10G   Yes
        0    port45  10G   Yes
        1    port25  10G   Yes
        1    port30  10G   Yes
        1    port34  10G   Yes
        1    port38  10G   Yes
        1    port42  10G   Yes
        1    port46  10G   Yes
        2    port28  10G   Yes
        2    port31  10G   Yes
        2    port35  10G   Yes
        2    port39  10G   Yes
        2    port43  10G   Yes
        2    port47  10G   Yes
        3    port27  10G   Yes
        3    port32  10G   Yes
        3    port36  10G   Yes
        3    port40  10G   Yes
        3    port44  10G   Yes
        3    port48  10G   Yes
------ ---- ------- ----- ----------
```

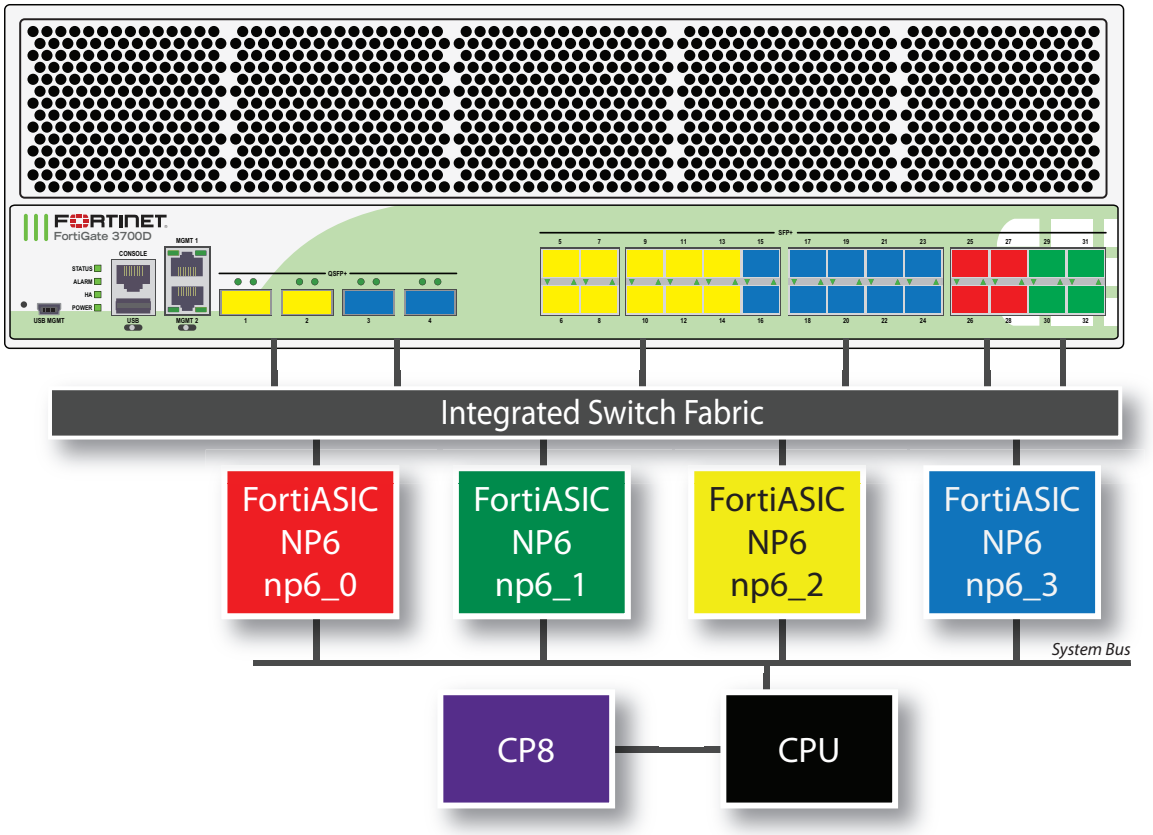
FortiGate-3700D fast path architecture

The FortiGate-3700D features four NP6 processors. By default the first two NP6 processors (np6_0 and np6_1) are configured for low latency operation. You can optionally disable low latency for these processors; which, also changes the FortiGate-3700D fast path architecture

FortiGate-3700D low latency fast path architecture

Ports 25 to 32 can be used for low latency offloading. As long as traffic enters and exits the FortiGate-3700D through ports connected to the same NP6 processor and using these low latency ports the traffic will be offloaded and have lower latency that other NP6 offloaded traffic. Latency is reduced by bypassing the integrated switch fabric. In low latency mode the FortiGate-3700D has the following fastpath architecture:

- Four SFP+ 10Gb interfaces, port25 through port28, share connections to the first NP6 processor (np6_0) so sessions entering one of these ports and exiting through another will experience low latency
- Four SFP+ 10Gb interfaces, port29 through port32, share connections to the second NP6 processor (np6_1) so sessions entering one of these ports and exiting through another will experience low latency
- Ten SFP+ 10Gb interfaces, port5 through port14, and two 40Gb QSFP interfaces, port1 and port2, share connections to the third NP6 processor (np6_2).
- Ten SFP+ 10Gb interfaces, port15 through port24, and two 40Gb QSFP interfaces, port3 and port4, share connections to the fourth NP6 processor (np6_3).



You can use the following command to display the low latency (default) FortiGate-3700D NP6 configuration. In this configuration, the first two NP6s (np6_0 and np6_1) are configured for low latency. The command output shows four NP6s named NP6_0, NP6_1, NP6_2, and NP6_3 and the interfaces (ports) connected to each NP6.

```
diag npu np6 port-list
Chip   XAUI Ports   Max   Cross-chip
                    Speed offloading
------ ---- ------- ----- ----------
np6_0  0    port25  10G   Yes
       1    port26  10G   Yes
       2    port27  10G   Yes
       3    port28  10G   Yes
------ ---- ------- ----- ----------
np6_1  0    port29  10G   Yes
       1    port30  10G   Yes
       2    port31  10G   Yes
       3    port32  10G   Yes
------ ---- ------- ----- ----------
np6_2  0    port5   10G   Yes
       0    port9   10G   Yes
       0    port13  10G   Yes
       1    port6   10G   Yes
       1    port10  10G   Yes
       1    port14  10G   Yes
       2    port7   10G   Yes
       2    port11  10G   Yes
       3    port8   10G   Yes
       3    port12  10G   Yes
       0-3  port1   40G   Yes
       0-3  port2   40G   Yes
------ ---- ------- ----- ----------
np6_3  0    port15  10G   Yes
       0    port19  10G   Yes
       0    port23  10G   Yes
       1    port16  10G   Yes
       1    port20  10G   Yes
       1    port24  10G   Yes
       2    port17  10G   Yes
       2    port21  10G   Yes
       3    port18  10G   Yes
       3    port22  10G   Yes
       0-3  port3   40G   Yes
       0-3  port4   40G   Yes
------ ---- ------- ----- ----------
```

FortiGate-3700D normal latency fast path architecture

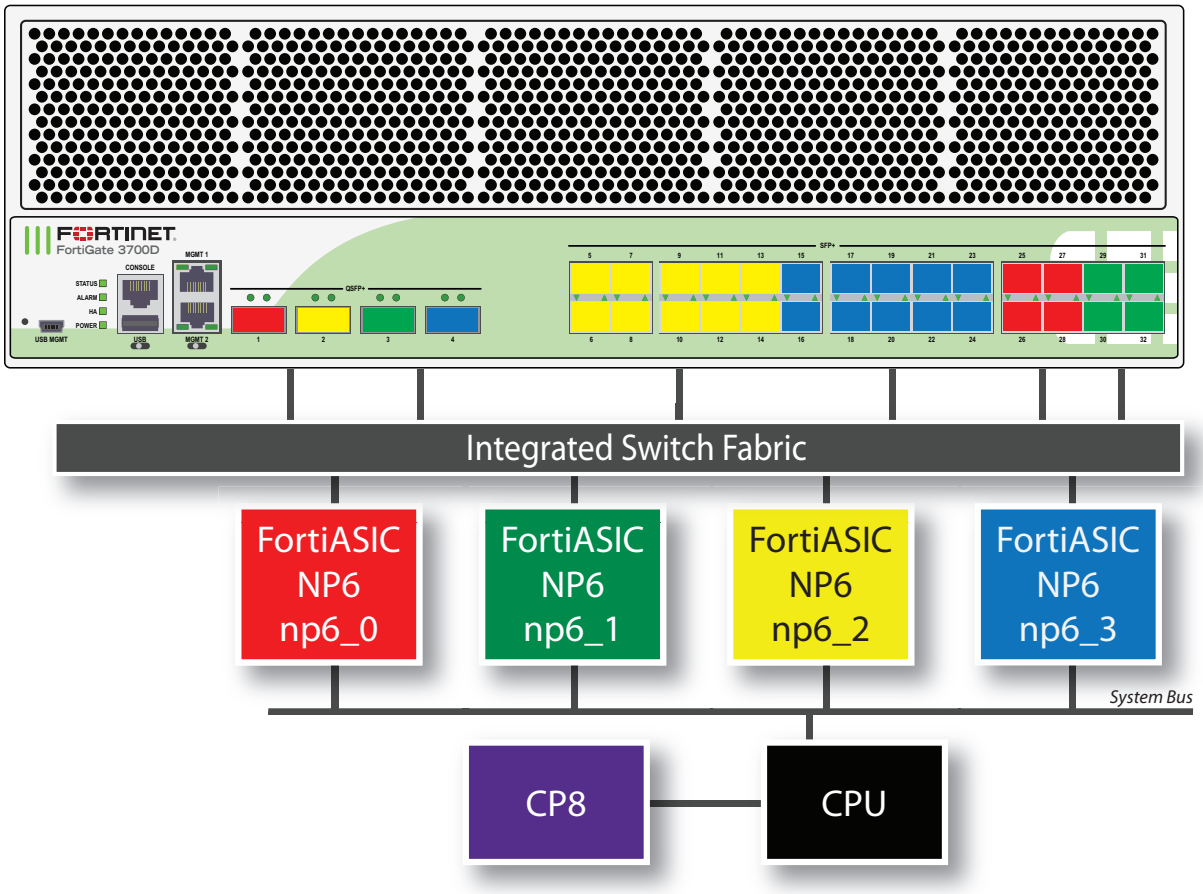
You can use the following command to turn off low latency mode for np6_0 and np6_1:

```
config system interface
   edit np6_0
      set low-latency-mode disable
   next
   edit np6_1
      set low-latency-mode disable
   end
```

You do not have to turn of low latency to both np6_0 and np6_1. If you turn off low latency to just one NP6, the other NP6 will still be mapped according to the default configuration.

In addition to turning off low latency, entering these commands also changes how ports are mapped to NP6s. Port1 is now mapped to np6_0 and port 3 is not mapped to np6_1. The FortiGate-3700D has the following fastpath architecture:

- One 40Gb QSFP interface, port1, and four SFP+ 10Gb interfaces, port25 through port28 share connections to the first NP6 processor (np6_0).
- One 40Gb QSFP interface, port3, and four SFP+ 10Gb interfaces, port29 through port32 share connections to the second NP6 processor (np6_1).
- One 40Gb QSFP interface, port2 and ten SFP+ 10Gb interfaces, port5 through port14 share connections to the third NP6 processor (np6_2).
- One 40Gb QSFP interface, port4, and ten SFP+ 10Gb interfaces, port15 through port24 share connections to the fourth NP6 processor (np6_3).



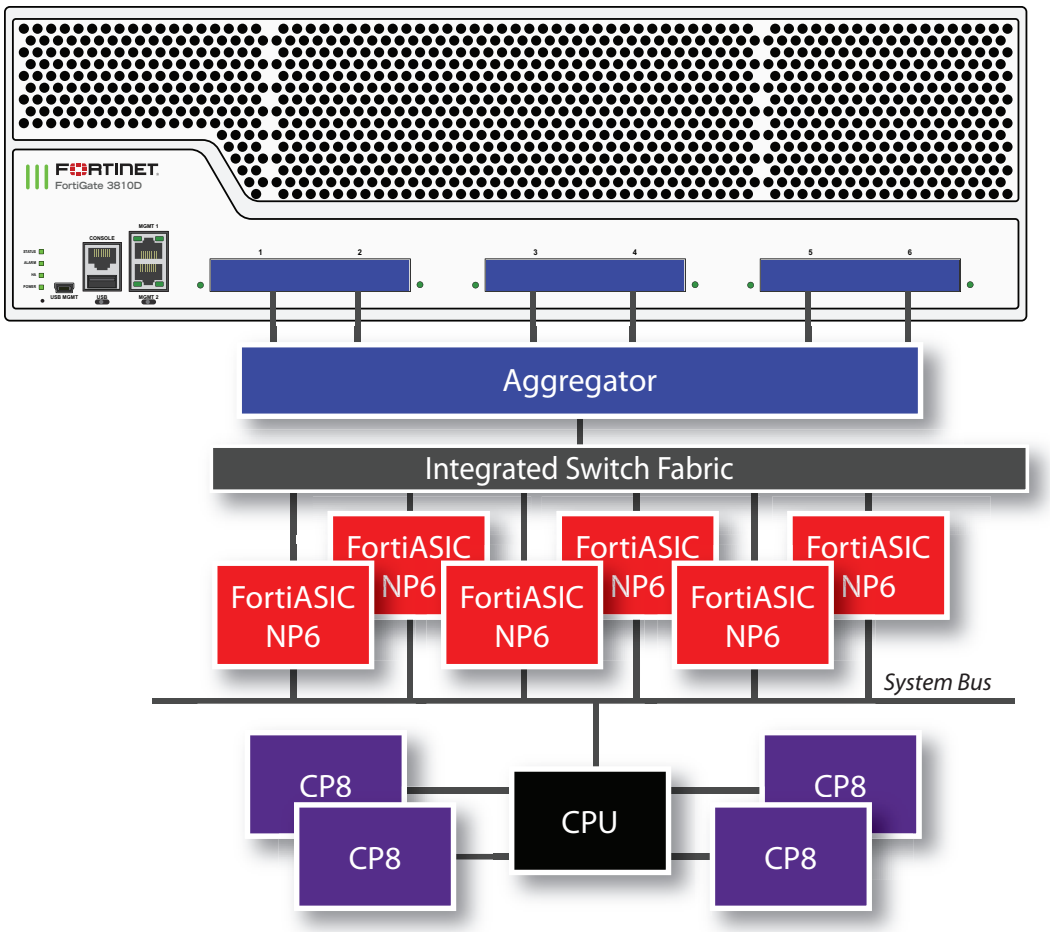
You can use the following command to display the FortiGate-3700D NP6 configuration with low latency turned off for np6_0 and np6_1. The command output shows four NP6s named NP6_0, NP6_1, NP6_2, and NP6_3 and the interfaces (ports) connected to each NP6.

```
diag npu np6 port-list
```

```
Chip   XAUI Ports   Max   Cross-chip
Speed offloading
------ ---- ------- ----- ----------
np6_0  0 port26     10G   Yes
       1    port25  10G   Yes
       2    port28  10G   Yes
       3    port27  10G   Yes
       0-3  port1   40G   Yes
------ ---- ------- ----- ----------
np6_1  0    port30  10G   Yes
       1    port29  10G   Yes
       2    port32  10G   Yes
       3    port31  10G   Yes
       0-3  port3   40G   Yes
------ ---- ------- ----- ----------
np6_2  0    port5   10G   Yes
       0    port9   10G   Yes
       0    port13  10G   Yes
       1    port6   10G   Yes
       1    port10  10G   Yes
       1    port14  10G   Yes
       2    port7   10G   Yes
       2    port11  10G   Yes
       3    port8   10G   Yes
       3    port12  10G   Yes
       0-3  port2   40G   Yes
------ ---- ------- ----- ----------
np6_3  0    port15  10G   Yes
       0    port19  10G   Yes
       0    port23  10G   Yes
       1    port16  10G   Yes
       1    port20  10G   Yes
       1    port24  10G   Yes
       2    port17  10G   Yes
       2    port21  10G   Yes
       3    port18  10G   Yes
       3    port22  10G   Yes
       0-3  port4   40G   Yes
------ ---- ------- ----- ----------
```

FortiGate-3810D fast path architecture

The FortiGate-3800D features 6 front panel 100Gigabit interfaces and eight NP6 processors connected to an Integrated Switch Fabirc (ISF). Individual interfaces are not mapped to NP6 processors. Instead an Aggregator connects the front panel interfaces to the ISF. Because of the aggregator, no special mapping is required for fastpath offloading.



You can use the following command to display the FortiGate-3810D NP6 configuration. The command output shows all NP6s connected to each interface (port) with cross-chip offloading supported for each port.

```
diag npu np6 port-list
Chip    XAUI Ports   Max      Cross-chip
                     Speed    offloading
------ ---- ------- ------   ----------
all     0-3  port1   100000M Yes
all     0-3  port2   100000M Yes
all     0-3  port3   100000M Yes
all     0-3  port4   100000M Yes
all     0-3  port5   100000M Yes
```

```
all     0-3  port6   100000M Yes
------ ---- ------- ------  ----------
```

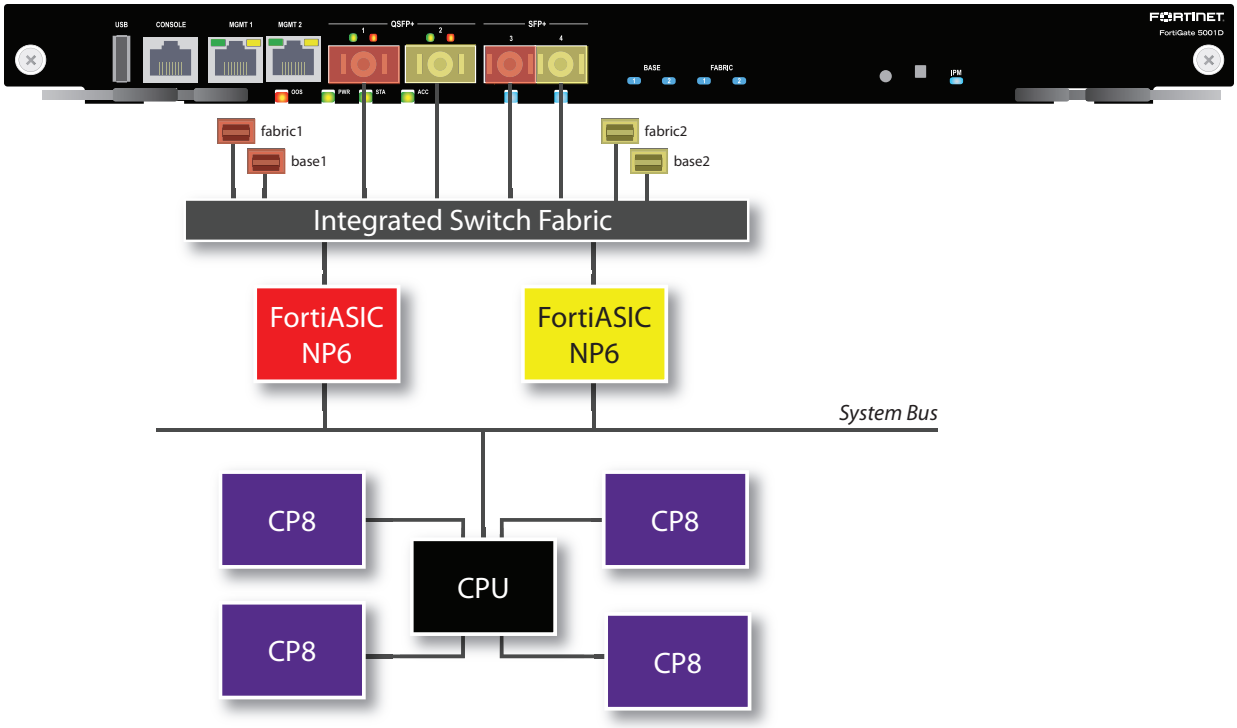
FortiGate-5001D fast path architecture

The FortiGate features two NP6 processors.

- port1, port3, fabric1 and base1 share connections to the first NP6 processor.
- port2, port4, fabric2 and base2 share connections to the second NP6 processor.

FortiGate NP6 to interface mapping

You can use the following command to display the FortiGate-5001D NP6 configuration. The command output shows two NP6s named NP6_0 and NP6_1. The output also shows the interfaces (ports) connected to each NP6.



```
diagnose npu np6 port-list
   Chip XAUI Ports       Max Cross-chip
                         Speed offloading
   ------ ---- --------    ----- ----------
   np6_0   0    elbc-ctrl/1 10G   Yes
           0    fctrl/f1-1  10G   Yes
           0    np6_0_29    10G   Yes
           0    fctrl/f3-1  10G   Yes
           0    np6_0_37    10G   Yes
           1    fctrl/f1-2  10G   Yes
           1    np6_0_30    10G   Yes
           1    fctrl/f3-2  10G   Yes
           1    np6_0_38    10G   Yes
           2    fctrl/f1-3  10G   Yes
           2    np6_0_31    10G   Yes
           3    fctrl/f3-3  10G   Yes
           2    np6_0_39    10G   Yes
           3    fctrl/f1-4  10G   Yes
           3    np6_0_32    10G   Yes
           3    fctrl/f3-4  10G   Yes
           3    np6_0_40    10G   Yes
           0-3  port1       40G   Yes
           0-3  port3       40G   Yes
           0-3  port4       40G   Yes
           0-3  base1       40G   Yes
           0-3  base2       40G   Yes
           0-3  np6_0_5     40G   Yes
           0-3  np6_0_6     40G   Yes
           0-3  fctrl/f1    40G   Yes
           0-3  np6_0_22    40G   Yes
           0-3  fctrl/f3    40G   Yes
           0-3  np6_0_24    40G   Yes
           0-3  np6_0_41    40G   Yes
           0-3  np6_0_42    40G   Yes
           0-3  np6_0_43    40G   Yes
   ------ ---- -------     ----  ----------
   np6_1   0    elbc-ctrl/2 10G   Yes
           0    np6_1_25    10G   Yes
           0    fctrl/f2-1  10G   Yes
           0    np6_1_33    10G   Yes
           0    fctrl/f4-1  10G   Yes
           1    np6_1_26    10G   Yes
           1    fctrl/f2-2  10G   Yes
           1    np6_1_34    10G   Yes
           1    fctrl/f4-2  10G   Yes
           2    np6_1_27    10G   Yes
           2    fctrl/f2-3  10G   Yes
           2    np6_1_35    10G   Yes
           2    fctrl/f4-3  10G   Yes
           3    np6_1_28    10G   Yes
           3    fctrl/f2-4  10G   Yes
           3    np6_1_36    10G   Yes
           3    fctrl/f4-4  10G   Yes
           0-3  port2       40G   Yes
           0-3  np6_1_0     40G   Yes
           0-3  np6_1_2     40G   Yes
           0-3  np6_1_3     40G   Yes
```

```
          0-3  np6_1_21    40G   Yes
          0-3  fctrl/f2    40G   Yes
          0-3  np6_1_23    40G   Yes
          0-3  fctrl/f4    40G   Yes
          0-3  np6_1_41    40G   Yes
          0-3  np6_1_42    40G   Yes
          0-3  np6_1_43    40G   Yes
------ ---- -------      ----- ----------
```

NP4 Acceleration

NP4 network processors provide fastpath acceleration by offloading communication sessions from the FortiGate CPU. When the first packet of a new session is received by an interface connected to an NP4 processor, just like any session connecting with any FortiGate interface, the session is forwarded to the FortiGate CPU where it is matched with a security policy. If the session is accepted by a security policy and if the session can be offloaded its session key is copied to the NP4 processor that received the packet. All of the rest of the packets in the session are intercepted by the NP4 processor and fast-pathed out of the FortiGate unit to their destination without ever passing through the FortiGate CPU. The result is enhanced network performance provided by the NP4 processor plus the network processing load is removed from the CPU. In addition, the NP4 processor can handle some CPU intensive tasks, like IPsec VPN encryption/decryption.

Session keys (and IPsec SA keys) are stored in the memory of the NP4 processor that is connected to the interface that received the packet that started the session. All sessions are fast-pathed and accelerated, even if they exit the FortiGate unit through an interface connected to another NP4. The key to making this possible is the Integrated Switch Fabric (ISF) that connects the NP4s and the FortiGate unit interfaces together. The ISF allows any port connectivity. All ports and NP4s can communicate with each other over the ISF.

There are no special ingress and egress fast path requirements because traffic enters and exits on interfaces connected to the same ISF. Most FortiGate models with multiple NP4 processors connect all interfaces and NP4 processors to the same ISF (except management interfaces) so this should not ever be a problem.

There is one limitation to keep in mind; the capacity of each NP4 processor. An individual NP4 processor has a capacity of 20 Gbps (10 Gbps ingress and 10 Gbps egress). Once an NP4 processor hits its limit, sessions that are over the limit are sent to the CPU. You can avoid this problem by as much as possible distributing incoming sessions evenly among the NP4 processors. To be able to do this you need to be aware of which interfaces connect to which NP4 processors and distribute incoming traffic accordingly.

Some FortiGate units contain one NP4 processor with all interfaces connected to it and to the ISF. As a result, offloading is supported for traffic between any pair of interfaces.

Some FortiGate units include NP4Lite processors. These network processors have the same functionality and limitations as NP4 processors but with about half the performance. NP4lite processors can be found in mid-range FortiGate models such as the FortiGate-200D and 240D.

Viewing your FortiGate's NP4 configuration

To list the NP4 network processors on your FortiGate unit, use the following CLI command.

```
get hardware npu np4 list
```

The output lists the interfaces that have NP4 processors. For example, for a FortiGate-5001C:

```
get hardware npu np4 list
ID    Model         Slot       Interface
0     On-board                 port1 port2 port3 port4
                               fabric1 base1 npu0-vlink0 npu0-vlink1
1     On-board                 port5 port6 port7 port8
                               fabric2 base2 npu1-vlink0 npu1-vlink1
```

NP4lite CLI commands (disabling NP4Lite offloading)

If your FortiGate unit includes an NP4Lite processor the following commands will be available:

- Use the following command to disable or enable NP4Lite offloading. By default NP4lite offloading is enabled. If you want to disable NP4Lite offloading to diagnose a problem enter:

```
diagnose npu nplite fastpath disable
```

This command disables NP4Lite offloading until your FortiGate reboots. You can also re-enable offloading by entering the following command:

```
diagnose npu nplite fastpath enable
```

- NP4lite debug command. Use the following command to debug NP4Lite operation:

```
diagnose npl npl_debug {<parameters>}
```

Configuring NP4 traffic offloading

Offloading traffic to a network processor requires that the FortiGate unit configuration and the traffic itself is suited to hardware acceleration. There are requirements for path the sessions and the individual packets.

NP4 session fast path requirements

Sessions must be fast path ready. Fast path ready session characteristics are:

- Layer 2 type/length must be 0x0800 (IEEE 802.1q VLAN specification is supported); link aggregation between any network interfaces sharing the same network processor(s) may be used (IEEE 802.3ad specification is supported)
- Layer 3 protocol must be IPv4
- Layer 4 protocol must be UDP, TCP or ICMP
- Layer 3 / Layer 4 header or content modification must not require a session helper (for example, SNAT, DNAT, and TTL reduction are supported, but application layer content modification is not supported)
- Firewall policies must not include proxy-based or flow-based security features (antivirus, web filtering, email filtering, IPS, application control, or DLP)
- Origin must not be local host (the FortiGate unit)

If you disable anomaly checks by Intrusion Prevention (IPS), you can still enable hardware accelerated anomaly checks using the `fp-anomaly` field of the `config system interface` CLI command. See Offloading NP4 anomaly detection on page 60

If a session is not fast path ready, the FortiGate unit will not send the session key to the network processor(s). Without the session key, all session key lookup by a network processor for incoming packets of that session fails, causing all session packets to be sent to the FortiGate unit's main processing resources, and processed at normal speeds.

If a session is fast path ready, the FortiGate unit will send the session key to the network processor(s). Session key lookup then succeeds for subsequent packets from the known session.

Packet fast path requirements

Packets within the session must then also meet packet requirements.

- Incoming packets must not be fragmented.
- Outgoing packets must not require fragmentation to a size less than 385 bytes. Because of this requirement, the configured MTU (Maximum Transmission Unit) for network processors' network interfaces must also meet or exceed the network processors' supported minimum MTU of 385 bytes.

If packet requirements are not met, an individual packet will use FortiGate unit main processing resources, regardless of whether other packets in the session are offloaded to the specialized network processor(s).

In some cases, due to these requirements, a protocol's session(s) may receive a mixture of offloaded and non-offloaded processing.

For example, FTP uses two connections: a control connection and a data connection. The control connection requires a session helper, and cannot be offloaded, but the data connection does not require a session helper, and can be offloaded. Within the offloadable data session, fragmented packets will not be offloaded, but other packets will be offloaded.

Some traffic types differ from general offloading requirements, but still utilize some of the network processors' encryption and other capabilities. Exceptions include IPsec traffic and active-active high availability (HA) load balanced traffic.

Mixing fast path and non-fast path traffic

If packet requirements are not met, an individual packet will be processed by the FortiGate CPU regardless of whether other packets in the session are offloaded to the NP4.

Also, in some cases, a protocol's session(s) may receive a mixture of offloaded and non-offloaded processing. For example, VoIP control packets may not be offloaded but VoIP data packets (voice packets) may be offloaded.

NP4 traffic shaping offloading

Accelerated Traffic shaping is supported with the following limitations.

- NP4 processors support policy-based traffic shaping. However, fast path traffic and traffic handled by the FortiGate CPU (slow path) are controlled separately, which means the policy setting on fast path does not consider the traffic on the slow path.
- The port based traffic policing as defined by the inbandwidth and outbandwidth CLI commands is not supported.
- DSCP configurations are supported.
- Per-IP traffic shaping is supported.
- QoS in general is not supported.

You can also use the traffic shaping features of the FortiGate unit's main processing resources by disabling NP4 offloding. See Disabling NP offloading for firewall policies on page 18.

NP4 IPsec VPN offloading

NP4 processors improve IPsec tunnel performance by offloading IPsec encryption and decryption.

Requirements for hardware accelerated IPsec encryption or decryption are a modification of general offloading requirements. Differing characteristics are:

- Origin can be local host (the FortiGate unit)
- In Phase 1 configuration, Local Gateway IP must be specified as an IP address of a network interface for a port attached to a network processor
- SA must have been received by the network processor
- in Phase 2 configuration:
  - encryption algorithm must be DES, 3DES, AES-128, AES-192, AES-256, or null
  - authentication must be MD5, SHA1, or null
  - if encryption is null, authentication must not also be null
  - if replay detection is enabled, `enc-offload-antireplay` must also be `enable` in the CLI

If replay detection is enabled in the Phase 2 configuration, you can enable or disable IPsec encryption and decryption offloading from the CLI. Performance varies by those CLI options and the percentage of packets requiring encryption or decryption. For details, see NP4 IPsec VPN offloading on page 52

To apply hardware accelerated encryption and decryption, the FortiGate unit's main processing resources must first perform Phase 1 negotiations to establish the security association (SA). The SA includes cryptographic processing instructions required by the network processor, such as which encryption algorithms must be applied to the tunnel. After ISAKMP negotiations, the FortiGate unit's main processing resources send the SA to the network processor, enabling the network processor to apply the negotiated hardware accelerated encryption or decryption to tunnel traffic.

Possible accelerated cryptographic paths are:

- IPsec decryption offload
  - Ingress ESP packet > Offloaded decryption > Decrypted packet egress (fast path)
  - Ingress ESP packet > Offloaded decryption > Decrypted packet to FortiGate unit's main processing resources
- IPsec encryption offload
  - Ingress packet > Offloaded encryption > Encrypted (ESP) packet egress (fast path)
  - Packet from FortiGate unit's main processing resources > Offloaded encryption > Encrypted (ESP) packet egress

NP4 IPsec VPN offloading configuration example

Hardware accelerated IPsec processing, involving either partial or full offloading, can be achieved in either tunnel or interface mode IPsec configurations.

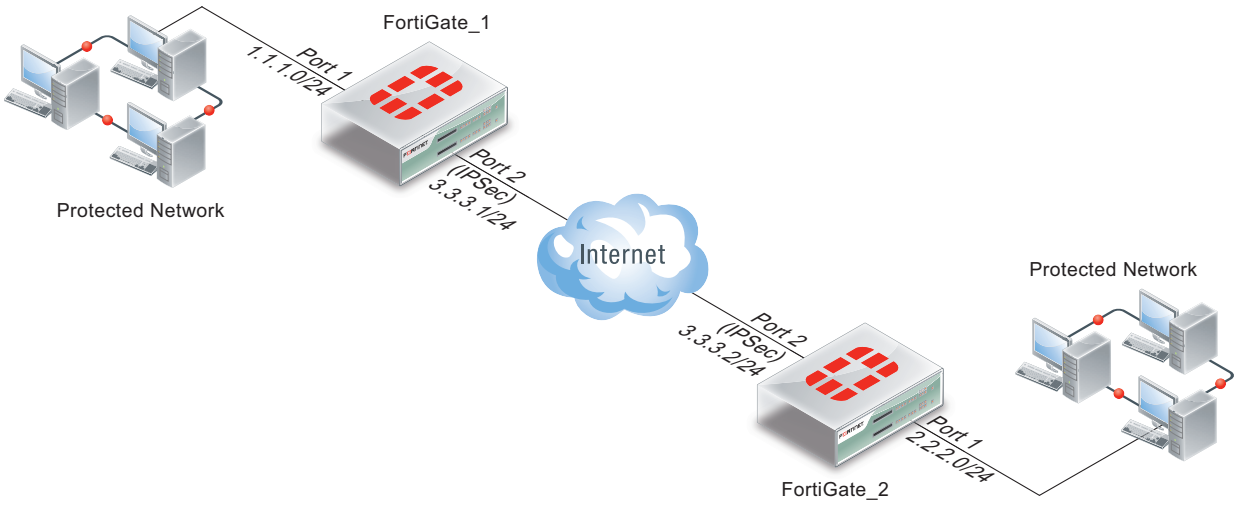
To achieve offloading for both encryption and decryption:

- In Phase 1 configuration's Advanced section, Local Gateway IP must be specified as an IP address of a network interface associated with a port attached to a network processor. (In other words, if Phase 1's Local Gateway IP is Main Interface IP, or is specified as an IP address that is not associated with a network interface associated with a port attached to a network processor, IPsec network processing is not offloaded.)
- In Phase 2 configuration's P2 Proposal section, if the checkbox "Enable replay detection" is enabled, `enc-offload-antireplay` and `dec-offload-antireplay` must be set to `enable` in the CLI.
- `offload-ipsec-host` must be set to `enable` in the CLI.

This section contains example IPsec configurations whose IPsec encryption and decryption processing is hardware accelerated by an NP4 unit contained in a FortiGate-5001B at both ends of the VPN tunnel.

Hardware accelerated IPsec VPN does not require both tunnel endpoints to have the same network processor model. However, if hardware is not symmetrical, the packet forwarding rate is limited by the slower side.

Example network topology for offloaded IPsec processing



Example ports and IP addresses for offloaded IPsec processing

	FortiGate_1		FortiGate_2	
	Port	**IP**	**Port**	**IP**
IPsec tunnel	FortiGate-5001B port 2	3.3.3.1/24	FortiGate-5001B port 2	3.3.3.2/24
Protected network	FortiGate-5001B port 1	1.1.1.0/24	FortiGate-5001B port 1	2.2.2.0/24

Accelerated policy mode IPsec configuration

The following steps create a hardware accelerated policy mode IPsec tunnel between two FortiGate-5001B units, each containing two NP4 processors, the first of which will be used.

To configure hardware accelerated policy mode IPsec

1. On FortiGate_1, go to **VPN > IPsec > Auto Key (IKE)**.
2. Configure Phase 1.
   For tunnel mode IPsec and for hardware acceleration, specifying the Local Gateway IP is required.
   Select **Advanced**. In the Local Gateway IP section, select **Specify** and type the VPN IP address 3.3.3.2, which is the IP address of FortiGate_2's FortiGate-ASM-FB4 module port 2.
3. Configure Phase 2.
4. Select **Enable replay detection**.
5. Use the following command to enable offloading antireplay packets:
   ```
   config system npu
      set enc-offload-antireplay enable
   end
   ```
   For details on encryption and decryption offloading options available in the CLI, see"Configuring NP accelerated VPN encryption/decryption offloading".
6. Go to **Policy > Policy > Policy**.
7. Configure a policy to apply the Phase 1 IPsec tunnel you configured in step 2 to traffic between FortiGate-5001B ports 1 and 2.
8. Go to **Router > Static > Static Route**.
9. Configure a static route to route traffic destined for FortiGate_2's protected network to VPN IP address of FortiGate_2's VPN gateway, 3.3.3.2, through the FortiGate-5001B port2.
   You can also configure the static route using the following CLI command:
   ```
   config router static
      edit 2
         set device "AMC-SW1/2"
         set dst 2.2.2.0 255.255.255.0
         set gateway 3.3.3.2
      end
   ```
10. On FortiGate_2, go to **VPN > IPsec > Auto Key (IKE)**.
11. Configure Phase 1.
    For tunnel mode IPsec and for hardware acceleration, specifying the Local Gateway IP is required.
    Select Advanced. In the Local Gateway IP section, select Specify and type the VPN IP address 3.3.3.1, which is the IP address of FortiGate_1's port2.
12. Configure Phase 2.
13. Select **Enable replay detection**.
14. Use the following command to enable offloading antireplay packets:
    ```
    config system npu
       set enc-offload-antireplay enable
    end
    ```
    For details on encryption and decryption offloading options available in the CLI, see "Configuring NP accelerated VPN encryption/decryption offloading".

15. Go to **Policy > Policy > Policy**.
16. Configure a policy to apply the Phase 1 IPsec tunnel you configured in step 9 to traffic between FortiGate-5001B ports 1 and 2.
17. Go to **Router > Static > Static Route**.
18. Configure a static route to route traffic destined for FortiGate_1's protected network to VPN IP address of FortiGate_1's VPN gateway, 3.3.3.1, through the FortiGate-5001B port2.
    You can also configure the static route using the following CLI commands:
    ```
    config router static
       edit 2
          set device "AMC-SW1/2"
          set dst 1.1.1.0 255.255.255.0
          set gateway 3.3.3.1
       end
    ```
19. Activate the IPsec tunnel by sending traffic between the two protected networks.
    To verify tunnel activation, go to **VPN > Monitor > IPsec Monitor**.

Accelerated interface mode IPsec configuration

The following steps create a hardware accelerated interface mode IPsec tunnel between two FortiGate units, each containing a FortiGate-ASM-FB4 module.

To configure hardware accelerated interface mode IPsec

1. On FortiGate_1, go to **VPN > IPsec > Auto Key (IKE)**.
2. Configure Phase 1.
   For interface mode IPsec and for hardware acceleration, the following settings are required.
   - Select Advanced.
   - Enable the checkbox "Enable IPsec Interface Mode."
   - In the Local Gateway IP section, select Specify and type the VPN IP address 3.3.3.2, which is the IP address of FortiGate_2's port 2.
3. Configure Phase 2.
4. Select **Enable replay detection**.
5. Use the following command to enable offloading antireplay packets:
   ```
   config system npu
      set enc-offload-antireplay enable
   end
   ```
   For details on encryption and decryption offloading options available in the CLI, see "Configuring NP accelerated VPN encryption/decryption offloading".
6. Go to **Policy > Policy > Policy**.
7. Configure two policies (one for each direction) to apply the Phase 1 IPsec configuration you configured in step 2 to traffic leaving from or arriving on FortiGate-ASM-FB4 module port 1.
8. Go to **Router > Static > Static Route**.
9. Configure a static route to route traffic destined for FortiGate_2's protected network to the Phase 1 IPsec device, `FGT_1_IPsec`.
   You can also configure the static route using the following CLI commands:
   ```
   config router static
      edit 2
   ```

```
      set device "FGT_1_IPsec"
      set dst 2.2.2.0 255.255.255.0
   end
```

10. On FortiGate_2, go to **VPN > IPsec > Auto Key (IKE)**.
11. Configure Phase 1.
    For interface mode IPsec and for hardware acceleration, the following settings are required.
    - Enable the checkbox “Enable IPsec Interface Mode.”
    - In the Local Gateway IP section, select Specify and type the VPN IP address 3.3.3.1, which is the IP address of FortiGate_1’s FortiGate-5001B port 2.
12. Configure Phase 2.
13. Select **Enable replay detection**.
14. Use the following command to enable offloading antireplay packets:

    ```
    config system npu
       set enc-offload-antireplay enable
    end
    ```

    For details on encryption and decryption offloading options available in the CLI, see "Hardware acceleration overview" on page 8.
15. Go to **Policy > Policy > Policy**.
16. Configure two policies (one for each direction) to apply the Phase 1 IPsec configuration you configured in step 9 to traffic leaving from or arriving on FortiGate-5001B port 1.
17. Go to **Router > Static > Static Route**.
18. Configure a static route to route traffic destined for FortiGate_1’s protected network to the Phase 1 IPsec device, `FGT_2_IPsec`.
    You can also configure the static route using the following CLI commands:

    ```
    config router static
       edit 2
          set device "FGT_2_IPsec"
          set dst 1.1.1.0 255.255.255.0
          next
       end
    ```

19. Activate the IPsec tunnel by sending traffic between the two protected networks.
    To verify tunnel activation, go to **VPN > Monitor > IPsec Monitor**.

Configuring Inter-VDOM link acceleration with NP4 processors

FortiGate units with NP4 processors include inter-VDOM links that can be used to accelerate inter-VDOM link traffic.

- For a FortiGate unit with two NP4 processors there are also two inter-VDOM links, each with two interfaces:
  - **npu0-vlink:**
    npu0-vlink0
    npu0-vlink1
  - **npu1-vlink:**
    npu1-vlink0
    npu1-vlink1

These interfaces are visible from the GUI and CLI. For a FortiGate unit with NP4 interfaces, enter the following CLI command (output shown for a FortiGate-5001B):

```
get hardware npu np4 list
ID    Model         Slot       Interface
0     On-board                 port1 port2 port3 port4
                               fabric1 base1 npu0-vlink0 npu0-vlink1
1     On-board                 port5 port6 port7 port8
                               fabric2 base2 npu1-vlink0 npu1-vlink1
```

By default the interfaces in each inter-VDOM link are assigned to the root VDOM. To use these interfaces to accelerate inter-VDOM link traffic, assign each interface in a pair to the VDOMs that you want to offload traffic between. For example, if you have added a VDOM named New-VDOM to a FortiGate unit with NP4 processors, you can go to **System > Network > Interfaces** and edit the **npu0-vlink1** interface and set the **Virtual Domain** to **New-VDOM**.

This results in an inter-VDOM link between root and New-VDOM. You can also do this from the CLI:

```
config system interface
   edit npu0-vlink1
      set vdom New-VDOM
end
```

Using VLANs to add more accelerated Inter-VDOM links

You can add VLAN interfaces to the accelerated inter-VDOM links to create inter-VDOM links between more VDOMs. For the links to work, the VLAN interfaces must be added to the same inter-VDOM link, must be on the same subnet, and must have the same VLAN ID.

For example, to accelerate inter-VDOM link traffic between VDOMs named Marketing and Engineering using VLANs with VLAN ID 100 go to **System > Network > Interfaces** and select **Create New** to create the VLAN interface associated with the Marketing VDOM:

Name	Marketing-link
Type	VLAN
Interface	npu0-vlink0
VLAN ID	100
Virtual Domain	Marketing
IP/Network Mask	172.20.120.12/24

Create the inter-VDOM link associated with Engineering VDOM:

Name	Engineering-link
Type	VLAN
Interface	npu0-vlink1
VLAN ID	100
Virtual Domain	Engineering
IP/Network Mask	172.20.120.22/24

Or do the same from the CLI:

```
config system interface
   edit Marketing-link
      set vdom Marketing
      set ip 172.20.120.12/24
      set interface npu0-vlink0
      set vlanid 100
   next
   edit Engineering-link
      set vdom Engineering
      set ip 172.20.120.22/24
      set interface npu0-vlink1
      set vlanid 100
```

Confirm that the traffic is accelerated

Use the following CLI commands to obtain the interface index and then correlate them with the session entries. In the following example traffic was flowing between new accelerated inter-VDOM links and physical ports port1 and port 2 also attached to the NP4 processor.

```
diagnose ip address list
IP=172.31.17.76->172.31.17.76/255.255.252.0 index=5 devname=port1
IP=10.74.1.76->10.74.1.76/255.255.252.0 index=6 devname=port2
IP=172.20.120.12->172.20.120.12/255.255.255.0 index=55 devname=IVL-VLAN1_ROOT
IP=172.20.120.22->172.20.120.22/255.255.255.0 index=56 devname=IVL-VLAN1_VDOM1

diagnose sys session list
session info: proto=1 proto_state=00 duration=282 expire=24 timeout=0 session info:
    proto=1 proto_state=00 duration=124 expire=59 timeout=0 flags=00000000
    sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=180/3/1 reply=120/2/1 tuples=2
orgin->sink: org pre->post, reply pre->post dev=55->5/5->55
    gwy=172.31.19.254/172.20.120.22
hook=post dir=org act=snat 10.74.2.87:768->10.2.2.2:8(172.31.17.76:62464)
hook=pre dir=reply act=dnat 10.2.2.2:62464->172.31.17.76:0(10.74.2.87:768)
misc=0 policy_id=4 id_policy_id=0 auth_info=0 chk_client_info=0 vd=0
serial=0000004e tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=10.74.2.87, bps=880
npu_state=00000000
npu info: flag=0x81/0x81, offload=4/4, ips_offload=0/0, epid=160/218, ipid=218/160,
    vlan=32769/0

session info: proto=1 proto_state=00 duration=124 expire=20 timeout=0 flags=00000000
    sockflag=00000000 sockport=0 av_idx=0 use=3
origin-shaper=
reply-shaper=
per_ip_shaper=
ha_id=0 policy_dir=0 tunnel=/
state=may_dirty npu
statistic(bytes/packets/allow_err): org=180/3/1 reply=120/2/1 tuples=2
orgin->sink: org pre->post, reply pre->post dev=6->56/56->6
    gwy=172.20.120.12/10.74.2.87
```

```
hook=pre dir=org act=noop 10.74.2.87:768->10.2.2.2:8(0.0.0.0:0)
hook=post dir=reply act=noop 10.2.2.2:768->10.74.2.87:0(0.0.0.0:0)
misc=0 policy_id=3 id_policy_id=0 auth_info=0 chk_client_info=0 vd=1
serial=0000004d tos=ff/ff ips_view=0 app_list=0 app=0
dd_type=0 dd_mode=0
per_ip_bandwidth meter: addr=10.74.2.87, bps=880
npu_state=00000000
npu info: flag=0x81/0x81, offload=4/4, ips_offload=0/0, epid=219/161, ipid=161/219,
    vlan=0/32769
total session 2
```

Offloading NP4 anomaly detection

Network interfaces associated with a port attached to an NP4 processor can be configured to offload anomaly checking to the NP4 processor. This anomaly checking happens before other offloading and separately from DoS policy anomaly checking. Using the following command, each FortiGate interface can have a different anomaly checking configuration even if they are connected to the same NP4 processor.

The options available for this command apply anomaly checking for NP4 sessions in the same way as the command described in Configuring individual NP6 processors on page 26 applies anomaly checking for for NP6 sessions.

```
config system interface
   edit <port-name>
      set fp-anomaly <anomalies>
   end
```

where `<anomalies>` can be one, more than one or all of the following:

Anomaly	Description
drop_icmp_frag	Drop ICMP fragments to pass.
drop_icmpland	Drop ICMP Land.
drop_ipland	Drop IP Land.
drop_iplsrr	Drop IP with Loose Source Record Route option.
drop_iprr	Drop IP with Record Route option.
drop_ipsecurity	Drop IP with Security option.
drop_ipssrr	Drop IP with Strict Source Record Route option.
drop_ipstream	Drop IP with Stream option.
drop_iptimestamp	Drop IP with Timestamp option.

Anomaly	Description
drop_ipunknown_option	Drop IP with malformed option.
drop_ipunknown_prot	Drop IP with Unknown protocol.
drop_tcp_fin_noack	Drop TCP FIN with no ACT flag set to pass.
drop_tcp_no_flag	Drop TCP with no flag set to pass.
drop_tcpland	Drop TCP Land.
drop_udpland	Drop UDP Land.
drop_winnuke	Drop TCP WinNuke.
pass_icmp_frag	Allow ICMP fragments to pass.
pass_icmpland	Allow ICMP Land to pass.
pass_ipland	Allow IP land to pass.
pass_iplsrr	Allow IP with Loose Source Record Route option to pass.
pass_iprr	Allow IP with Record Route option to pass.
pass_ipsecurity	Allow IP with Security option to pass.
pass_ipssrr	Allow IP with Strict Source Record Route option to pass.
pass_ipstream	Allow IP with Stream option to pass.
pass_iptimestamp	Allow IP with Timestamp option to pass.
pass_ipunknown_option	Allow IP with malformed option to pass.
pass_ipunknown_prot	Allow IP with Unknown protocol to pass.
pass_tcp_fin_noack	Allow TCP FIN with no ACT flag set to pass.
pass_tcp_no_flag	Allow TCP with no flag set to pass.
pass_tcpland	Allow TCP Land to pass.

Anomaly	Description
pass_udpland	Allow UDP Land to pass.
pass_winnuke	Allow TCP WinNuke to pass.

Example

You might configure an NP4 to drop packets with TCP WinNuke or unknown IP protocol anomalies, but to pass packets with an IP time stamp, using hardware acceleration provided by the network processor.

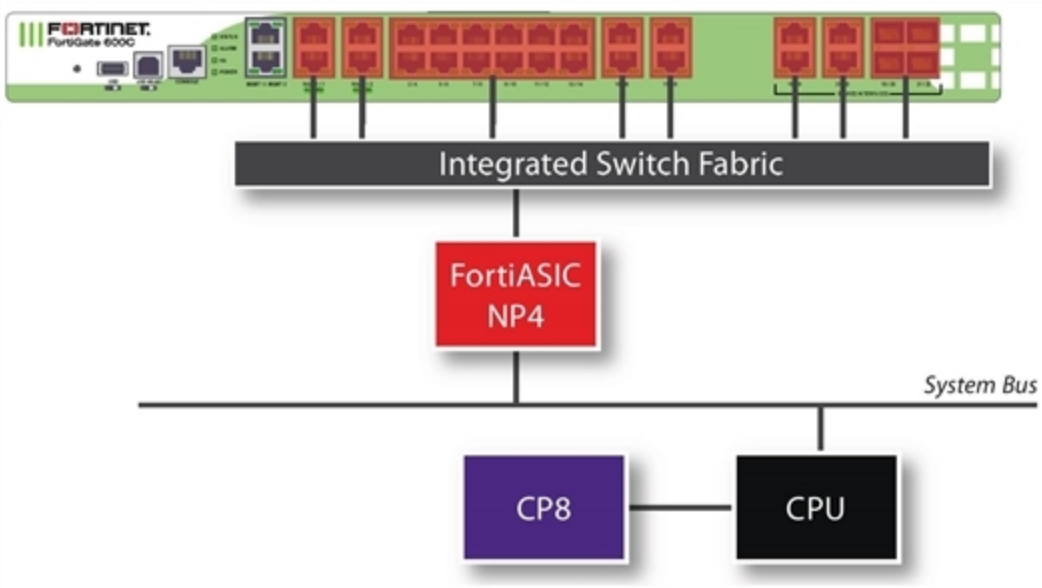
```
config system interface
   edit port1
      set fp-anomaly drop_winnuke drop_ipunknown_prot pass_iptimestamp
   end
```

FortiGate NP4 architectures

This chapter shows the NP4 architecture for the all FortiGate units and modules that include NP4 processors.

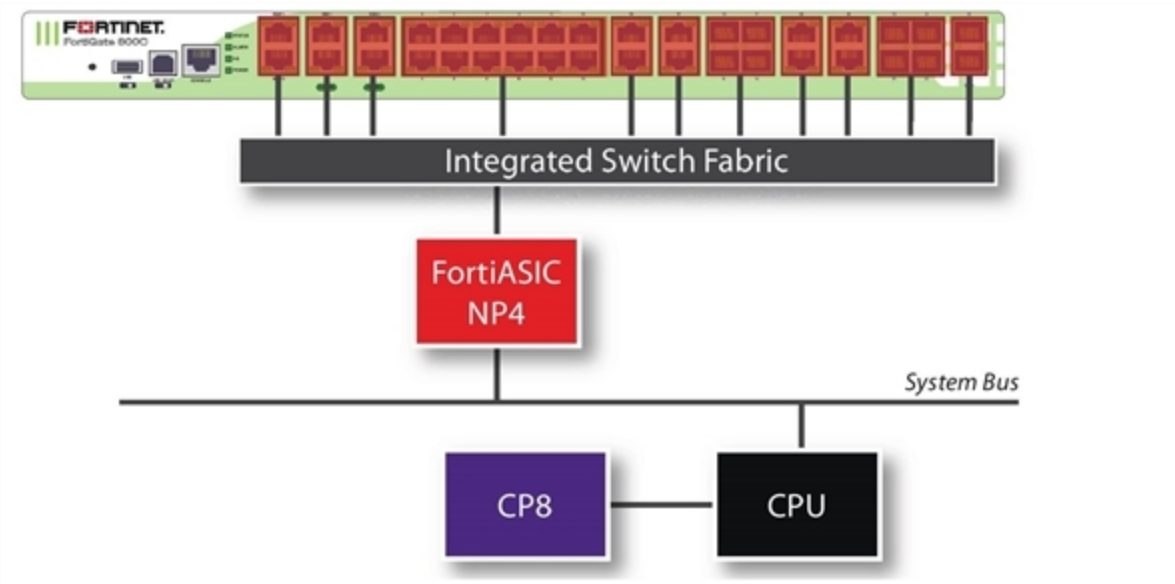
FortiGate-600C

The FortiGate-600C features one NP4 processor. All the ports are connected to this NP4 over the Integrated Switch Fabric. Port1 and port2 are dual failopen redundant RJ-45 ports. Port3-port22 are RJ-45 ethernet ports, and there are four 1Gb SFP interface ports duplicating the port19-port22 connections.



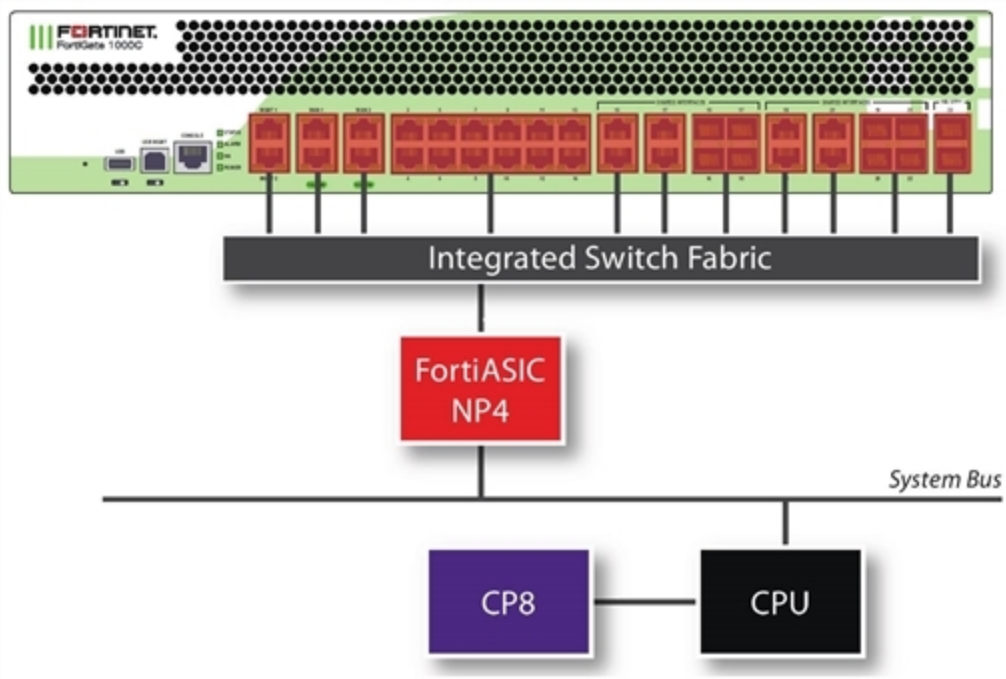
FortiGate-800C

The FortiGate-800C features one NP4 processor. All the ports are connected to this NP4. Port1 and port2 are dual failopen redundant RJ-45 ports. Port3-port22 are RJ-45 ethernet ports, and there are eight 1Gb SFP interface ports duplicating the port15-18 and port19-port22 connections. There are also two 10Gb SFP+ ports, port23 and port24.



FortiGate-1000C

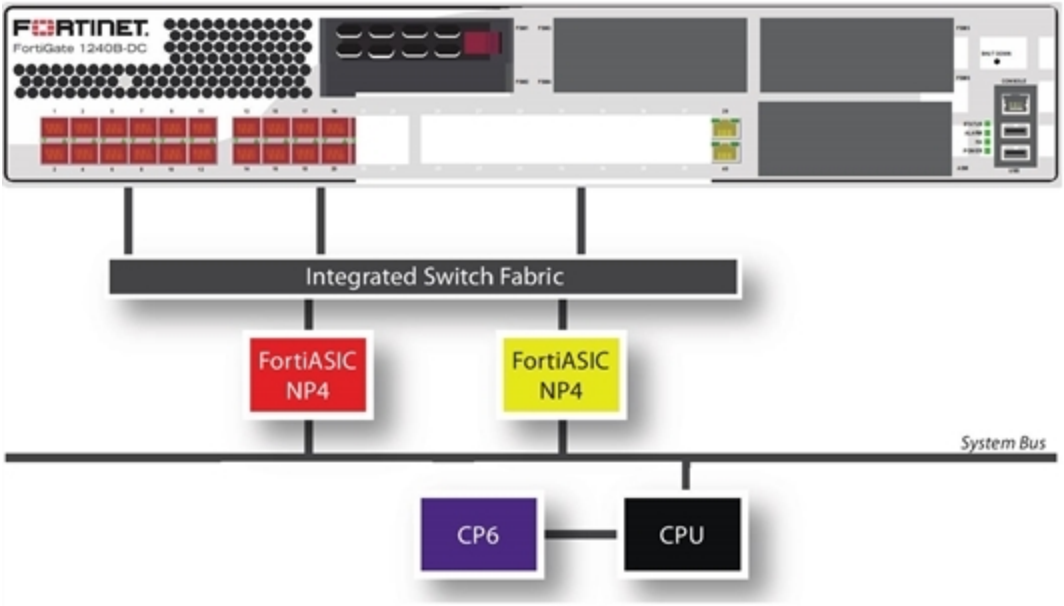
The FortiGate-1000C features one NP4 processor. All the ports are connected to this NP4. Port1 and port2 are dual failopen redundant RJ-45 ports. Port3-port22 are RJ-45 ethernet ports, and there are eight 1Gb SFP interface ports duplicating the port15-18 and port19-port22 connections. There are also two 10Gb SFP+ ports, port23 and port24.



FortiGate-1240B

The FortiGate-1240B features two NP4 processors:

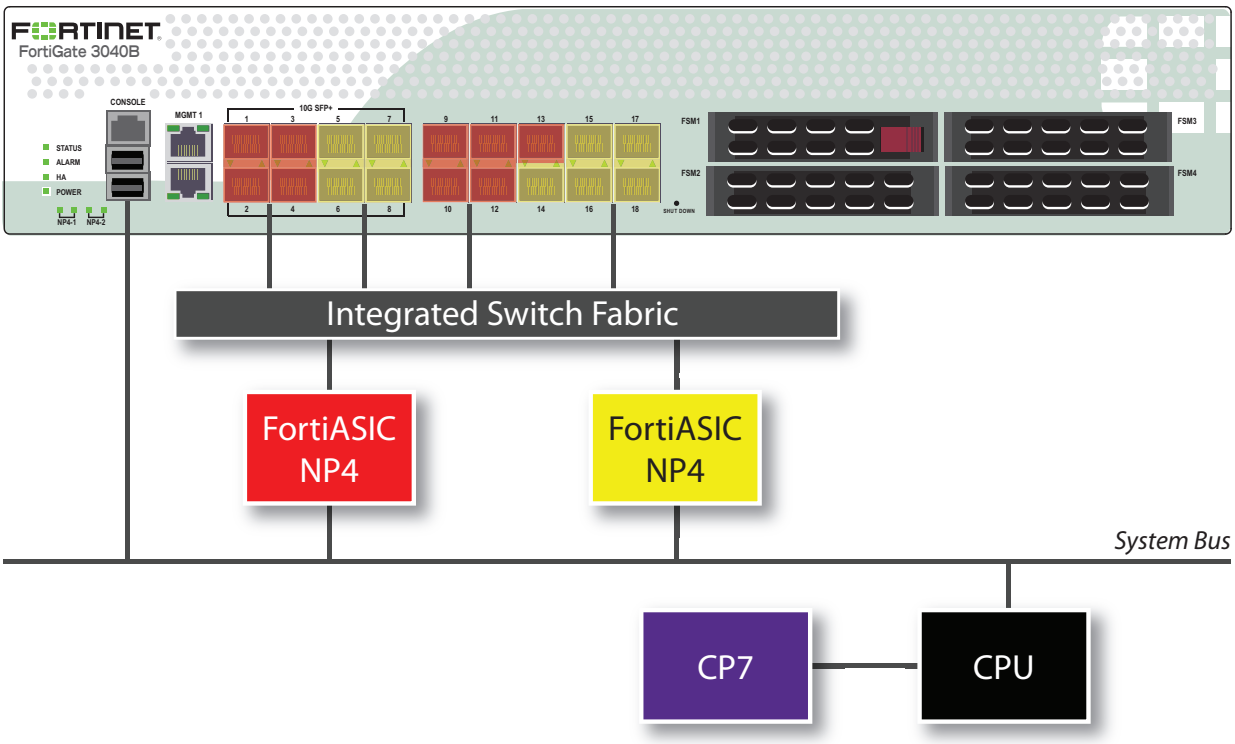
- Port1-port24 are 1Gb SFP interfaces connected to one NP4 processor.
- Port25-port40 are RJ-45 ethernet ports, connected to the other NP4 processor.



FortiGate-3040B

The FortiGate-3040B features two NP4 processors:

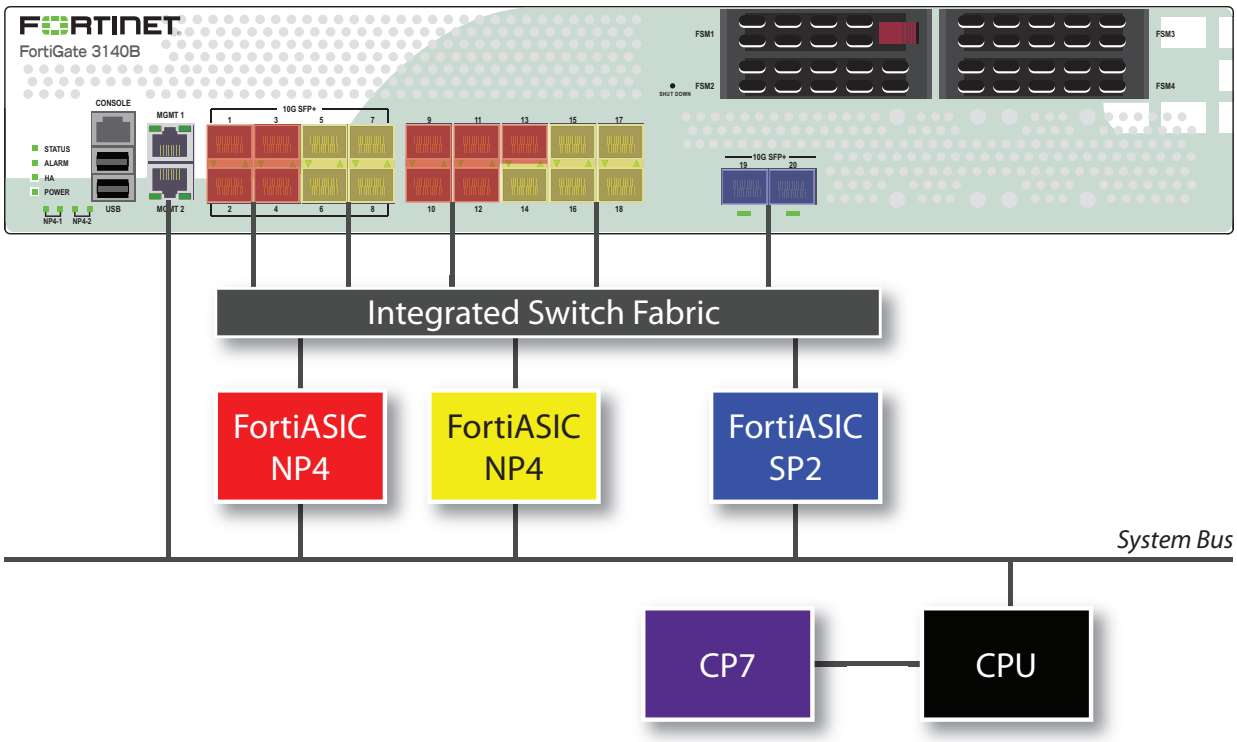
- The 10Gb interfaces, port1, port2, port3, port4, and the 1Gb interfaces, port9, port10, port11, port12, port13, share connections to one NP4 processor.
- The 10Gb interfaces, port5, port6, port7, port8, and the 1Gb interfaces, port14, port15, port16, port17, port18, share connections to the other NP4 processor.



FortiGate-3140B

The FortiGate-3140B features two NP4 processors and one SP2 processor:

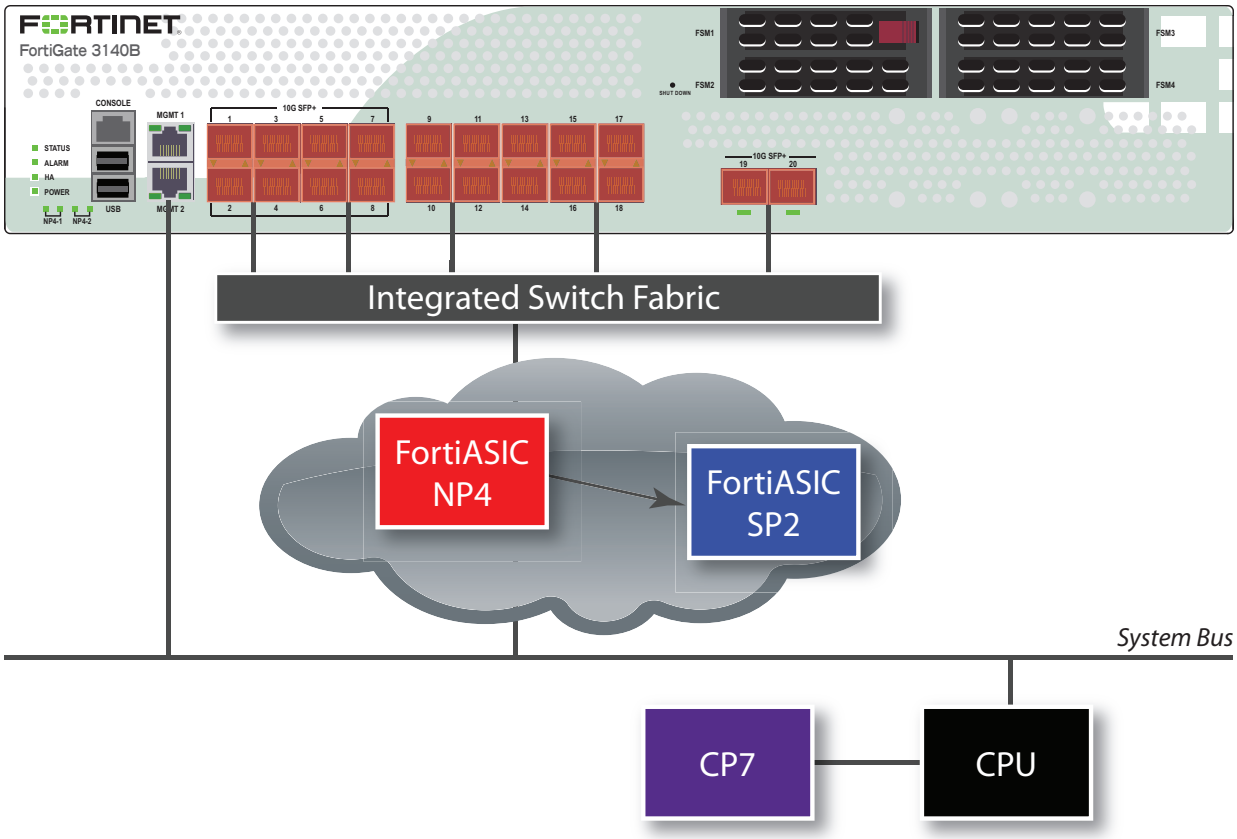
- The 10Gb interfaces, port1, port2, port3, port4, and the 1Gb interfaces, port9, port10, port11, port12, port13, share connections to one NP4 processor.
- The 10Gb interfaces, port5, port6, port7, port8, and the 1Gb interfaces, port14, port15, port16, port17, port18, share connections to the other NP4 processor.
- The 10Gb interfaces, port19 and port20, share connections to the SP2 processor.



FortiGate-3140B — load balance mode

The FortiGate-3140B load balance mode allows you increased flexibility in how you use the interfaces on the FortiGate unit. When enabled, traffic between any two interfaces (excluding management and console) is accelerated. Traffic is not limited to entering and leaving the FortiGate unit in specific interface groupings to benefit from NP4 and SP2 acceleration. You can use any pair of interfaces.

Security acceleration in this mode is limited, however. Only IPS scanning is accelerated in load balance mode.



To enable this feature, issue this CLI command.

```
config system global
   set sp-load-balance enable
end
```

The FortiGate unit will then restart.

To return to the default mode, issue this CLI command.

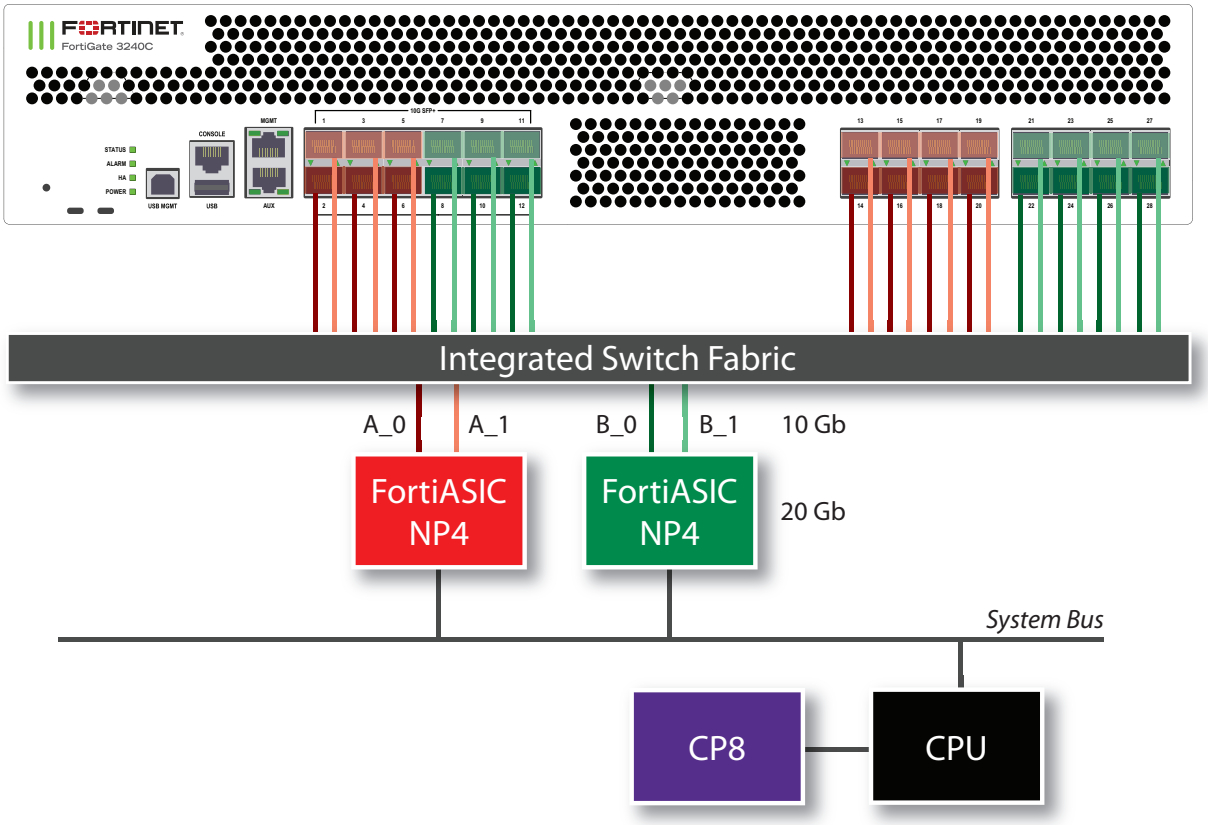
```
config system global
   set sp-load-balance disable
end
```

FortiGate-3240C

The FortiGate-3240C features two NP4 processors:

- The 10Gb interfaces, port1 through port6, and the 1Gb interfaces, port13 through port20, share connections to one NP4 processor.
- The 10Gb interfaces, port7 through port12, and the 1Gb interfaces, port21 through port28, share connections to the other NP4 processor.

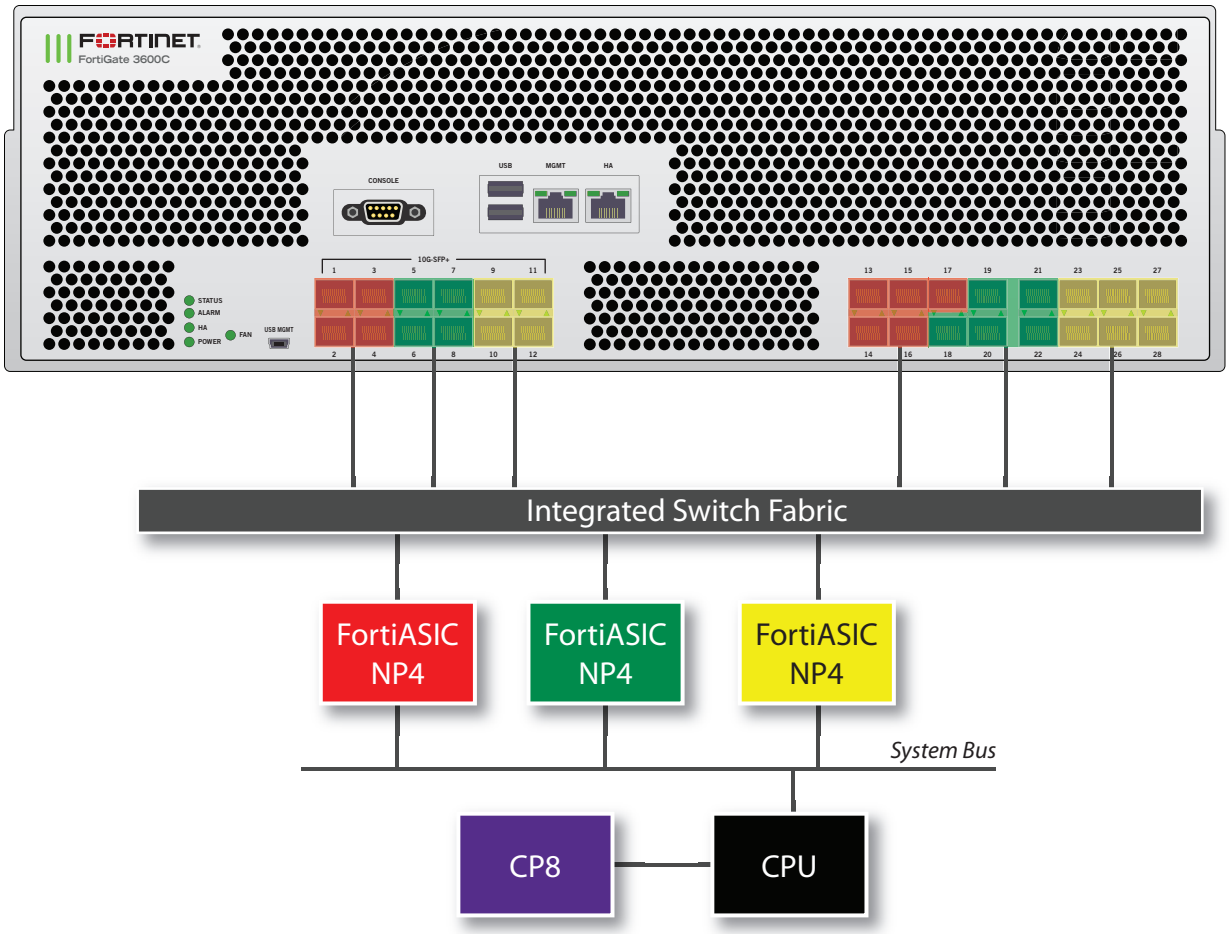
In addition to the ports being divided between the two NP4 processors, they are further divided between the two connections to each processor. Each NP4 can process 20 Gb of network traffic per second and each of two connections to each NP4 can move 10Gb of data to the processor per second, so the ideal configuration would have no more than 10 Gb of network traffic to each connection of each NP4 at any time.



FortiGate-3600C

The FortiGate-3600C features three NP4 processors:

- The 10Gb interfaces, port1-port4, and the 1Gb interfaces, port13-port17, share connections to one NP4 processor.
- The 10Gb interfaces, port5-port8, and the 1Gb interfaces, port18-port22 share connections to the second NP4 processor.
- The 10Gb interfaces, port9-port12, and the 1Gb interfaces, port23-port28 share connections to the third NP4 processor.



XAUI interfaces

Each NP4 processor connects to the integrated switch fabric through two XAUI interfaces: XAUI0 and XAUI1. On each NP4 processor all of the odd numbered interfaces use XAU0 and all of the even numbered interfaces use XAUI1:

NPU1

XAUI0 = port1,port3,port13, port15, port17

XAUI1 = port2, port4, port14, port16

NPU2

XAUI0 = port5, port7, port18, port20, port22

XAUI1 = port6, port8, port19, port21

NPU3

XAUI0 = port9, port11, port23, port25, port27

XAUI1 = port10, port12, port24, port26, port28

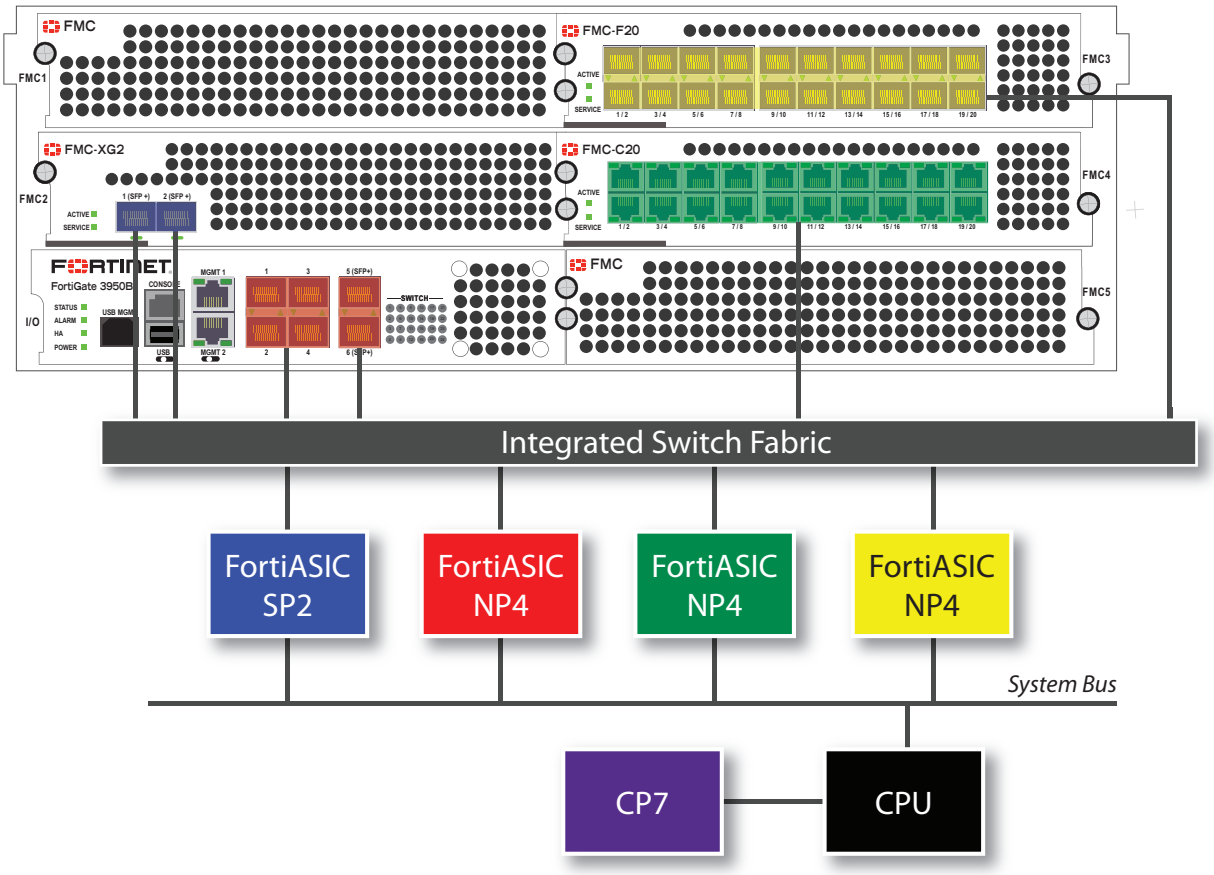
Usually you do not have to be concerned about the XAUI interface mapping. However, if an NP4 interface is processing a very high amount of traffic you should distribute that traffic among both of the XAUI interfaces connected to it. So if you have a very high volume of traffic flowing between two networks you should connect both networks to the same NP4 processor but to different XAUI links. So between even and an add numbered FortiGate-3600C ports. For example, you could connect one network to port5 and the other network to port6. In this configuration, the second NP4 processor would handle traffic acceleration and both XAUI interfaces would be processing traffic.

FortiGate-3950B and FortiGate-3951B

The FortiGate-3950B features one NP4 processor. The 1Gb SPF interfaces, port1, port2, port3, port4, and the 10Gb SPF+ interfaces, port5, port6, share connections to one NP4 processor. The FortiGate-3951B is similar to the FortiGate-3950B, except it trades one FMC slot for four FSM slots. The network interfaces available on each model are identical.

You can add additional FMC interface modules. The diagram below shows a FortiGate-3950B with three modules installed: an FMC-XG2, an FMC-F20, and an FMC-C20.

- The FMC-XG2 has one SP2 processor. The 10Gb SPF+ interfaces, port1 and port2, share connections to the processor.
- The FMC-F20 has one NP4 processor and the twenty 1Gb SPF interfaces, port1 through port20, share connections to the NP4 processor.
- The FMC-C20 has one NP4 processor and the twenty 10/100/1000 interfaces, port1 through port20, share connections to the NP4 processor.


FMC
FMC1
FMC-F20
FMC3
FMC-XG2
FMC2
FMC-C20
FMC4
FMC
FMC5
Integrated Switch Fabric
FortiASIC SP2
FortiASIC NP4
FortiASIC NP4
FortiASIC NP4
System Bus
CP7
CPU

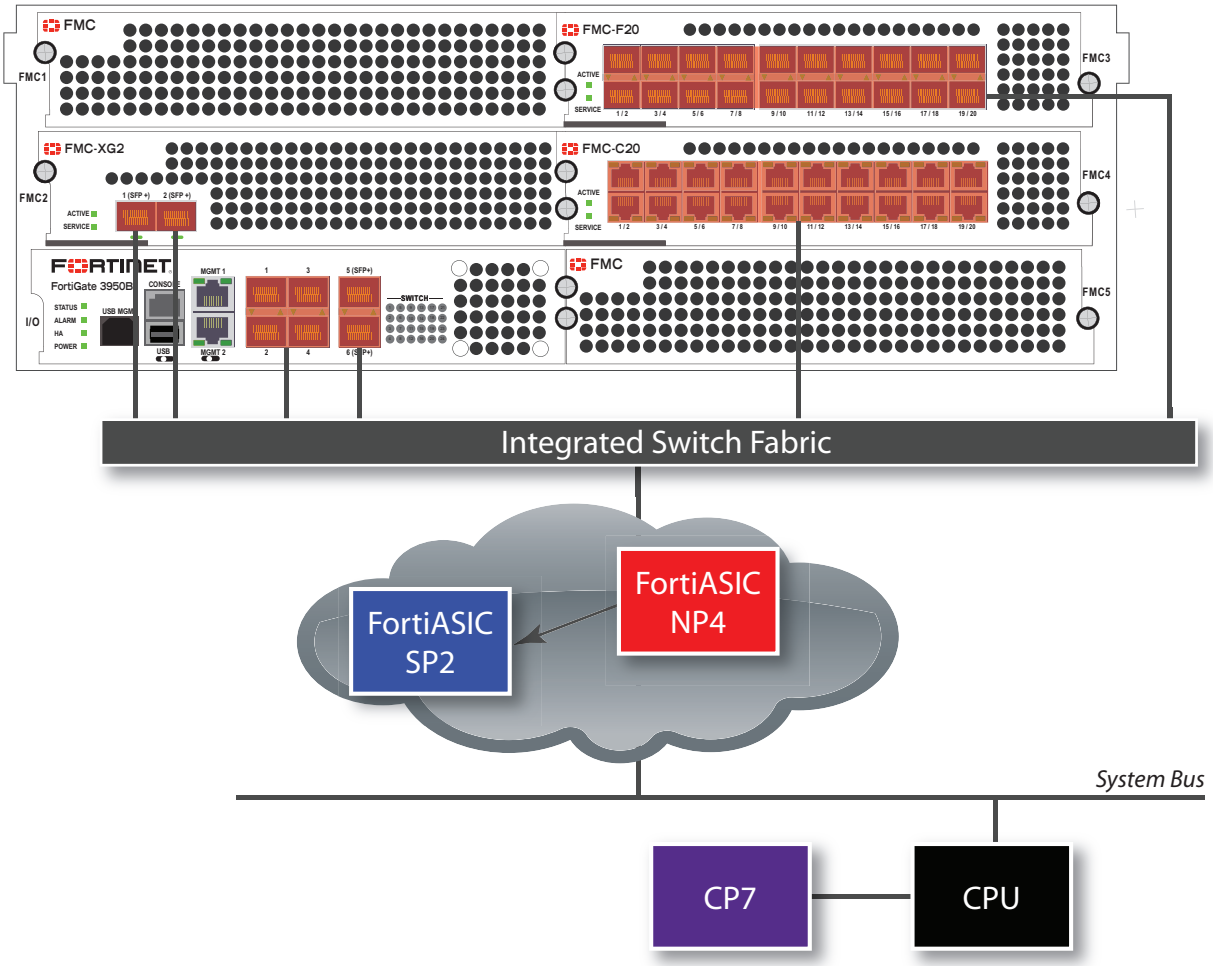
FortiGate-3950B and FortiGate-3951B — load balance mode

Adding one or more FMC-XG2 modules to your FortiGate-3950B allows you to enable load balance mode. This feature allows you increased flexibility in how you use the interfaces on the FortiGate unit. The FortiGate-3951B is similar to the FortiGate-3950B, except it trades one FMC slot for four FSM slots. The network interfaces available on each model are identical.

When enabled, traffic between any two interfaces (excluding management and console) is accelerated whether they are the six interfaces on the FortiGate-3950B itself, or on any installed FMC modules. Traffic is not limited to entering and leaving the FortiGate unit in specific interface groupings to benefit from NP4 and SP2 acceleration. You can use any pair of interfaces.

Security acceleration in this mode is limited, however. Only IPS scanning is accelerated in load balance mode.

The FortiGate-3950B in load balance mode



To enable this feature, issue this CLI command.

```
config system global
   set sp-load-balance enable
```

```
    end
```

The FortiGate unit will then restart.

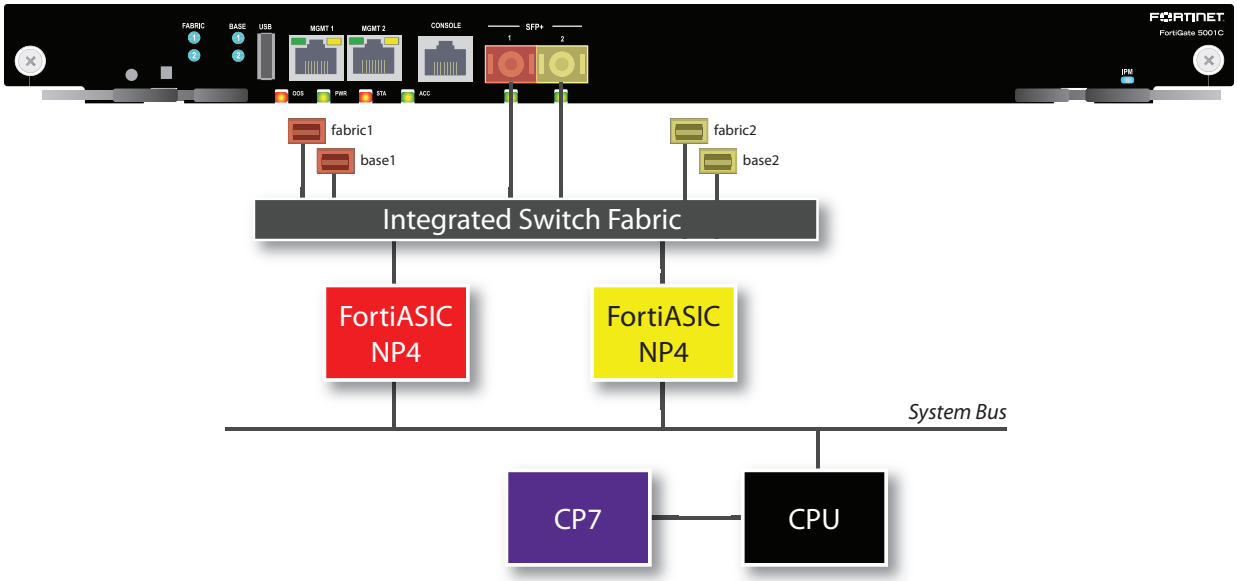
To return to the default mode, issue this CLI command.

```
config system global
   set sp-load-balance disable
end
```

FortiGate-5001C

The FortiGate-5001C board includes two NP4 processors connected to an integrated switch fabric:

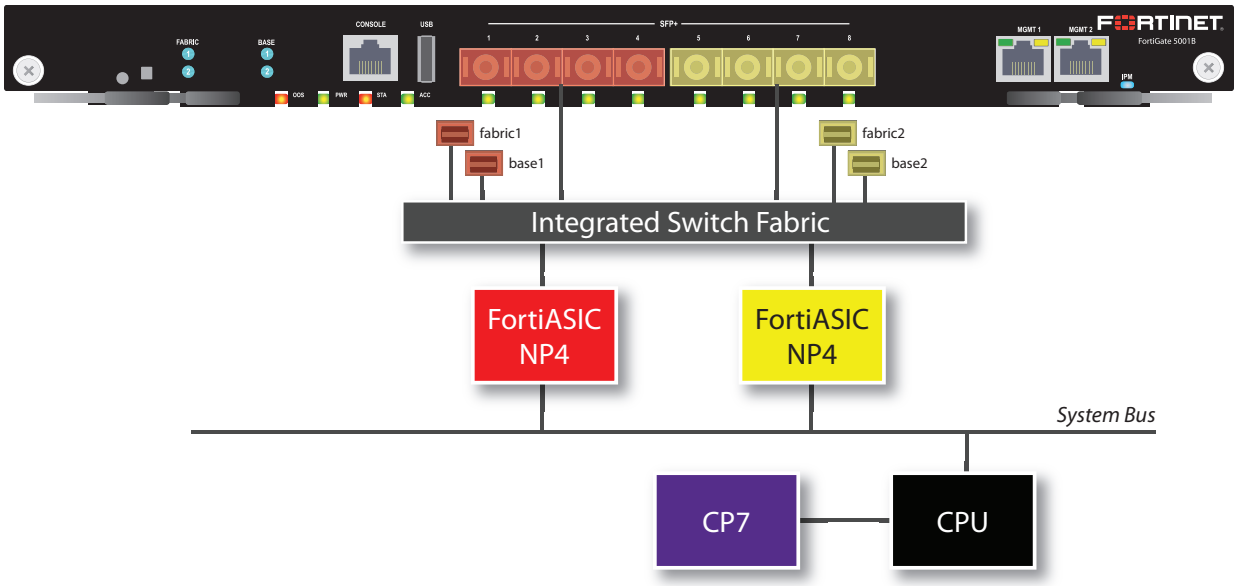
- The port1, fabric1, and base1 interfaces are connected to one NP4 processor.
- The port2, fabric2, and base2 interfaces are connected to the other NP4 processor.



FortiGate-5001B

The FortiGate-5001B board includes two NP4 connected to an integrated switch fabric.

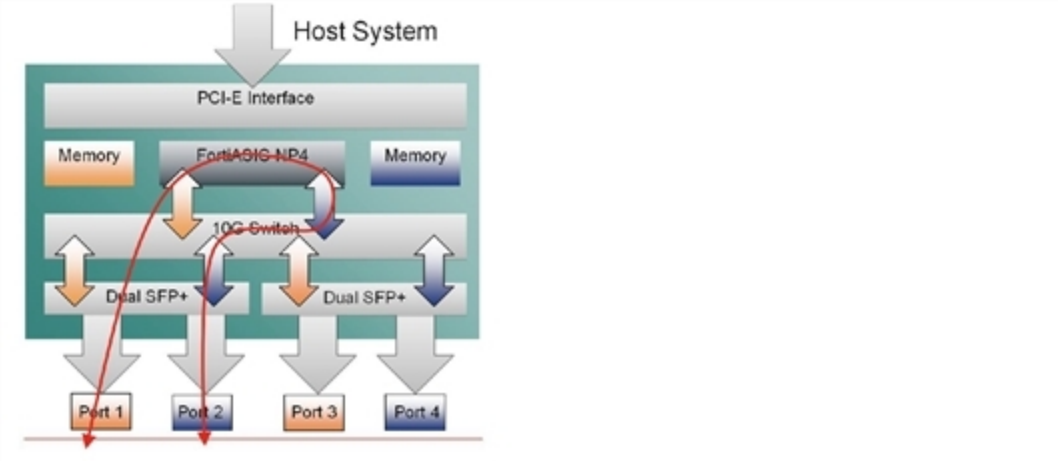
- The port1, port2, port3, port4, fabric1 and base1 interfaces are connected to one NP4 processor.
- The port5, port6, port7, port8, fabric2 and base2 interfaces are connected to the other NP4 processor.



Setting switch-mode mapping on the ADM-XD4

The ADM-XD4 SP has four 10Gb/s ports, but the NP4 processor it contains has only two 10Gb/s ports. The external ports you use are important to optimize the SP for your application.

ADM-XD4 mapping mode



Ports 1 and 3 share one NP4 processor and ports 2 and 4 share the other. Performance ports sharing the same NP4 processor is far better than when forcing network data to move between NP4 processors by using one port from each, for example ports 1 and 2 or ports 3 and 4.